NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DEVELOPMENT OF A DYNAMIC MODEL FOR A UAV

by

Evangelos C. Papageorgiou

March 1997

Thesis Advisor: Isaac Kaminer

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE March 1997	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE Development Of A Dynamic Model For A UAV

6. AUTHOR(S) Evangelos C. Papageorgiou

5. FUNDING NUMBERS

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited	12b. DISTRIBUTION CODE

13. ABSTRACT
Moments of inertia were experimentally determined and the longitudinal and lateral/directional static and dynamic static and dynamic stability and control derivatives were estimated for a fixed wing Unmanned Air Vehicle (UAV). High fidelity, non-linear equations of motion were derived and tailored for use on the specific aircraft. Computer modeling of these resulting equations was employed both in Matlab/Simulink and in Matrix$_x$/Systembuild. The resulting computer model was linearized at a specific flight condition, and the dynamics of the aircraft were predicted. Several flight tests were conducted at a nearby airfield and the behavior of the aircraft was compared to that of the computer model. The longitudinal dynamics as depicted by the short period mode were found to be almost identical with those predicted by the non-linear computer model. The phugoid mode was also observed and found to be in close agreement. In the lateral/directional dynamics, flight test was employed to improve the model and the parameters were modified to obtain a better match. Ultimately a reasonably accurate non-linear model was achieved as required for purposes of control and navigation system design.

14. SUBJECT TERMS Unmanned Aerial Vehicle, UAV, Flight Dynamics, Frog, Matrix$_x$, Systembuild, Parameter Estimation, Matlab, Simulink, Flight Test	15. NUMBER OF PAGES 116
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

DEVELOPMENT OF A DYNAMIC MODEL FOR A UAV

Evangelos C. Papageorgiou
Lieutenant, Hellenic Navy
B.S. Hellenic Naval Academy, 1988

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1997

ABSTRACT

Moments of inertia were experimentally determined and the longitudinal and lateral/directional static and dynamic stability and control derivatives were estimated for a fixed wing Unmanned Air Vehicle (UAV). High fidelity, non-linear equations of motion were derived and tailored for use on the specific aircraft. Computer modeling of these resulting equations was employed both in Matlab/Simulink and in $Matrix_x$/Systembuild. The resulting computer model was linearized at a specific flight condition, and the dynamics of the aircraft were predicted. Several flight tests were conducted at a nearby airfield and the behavior of the aircraft was compared to that of the computer model. The longitudinal dynamics as depicted by the short period mode were found to be almost identical with those predicted by the nonlinear computer model. The phugoid mode was also observed and found to be in close agreement. In the lateral/directional dynamics, flight test was employed to improve the model and the parameters were modified to obtain a better match. Ultimately a reasonably accurate nonlinear model was achieved as required for purposes of control and navigation system design.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

I would like to express my appreciation to the staff and faculty of the Naval Postgraduate School for making my education an outstanding experience. A very special thanks to all the people who contributed to this thesis in their own individual way.

I want to thank in particular Dr. I. I. Kaminer for his guidance, his boundless patience, and his valuable teaching. He provided me with the motivation to get through this project and made it a worthwhile experience. I want to express my appreciation to Dr. R. M. Howard. His professional counsel and guidance was indeed invaluable and has been a driving factor towards the completion of this work.

I would to like to thank LCdr. Eric Hallberg for his significant help and for developing the tools necessary for an outstanding avionics laboratory. He has been one of the persons who contributed in the success of the flight tests. Also a very special thanks to Mr. Don Meeks for his help in my experimental work and his remarkable abilities as a pilot during the flight tests.

I. INTRODUCTION

A. MISSION NEED

Unmanned Aerial Vehicles (UAVs) are becoming an increasingly important part of the armed forces operations. During various military missions the UAVs have the capability to collect intelligence and target for gunfire support, gather battle damage assessment, as well as perform many other tasks. The real benefit in using unmanned aircraft lies in the fact that no lives are jeopardized when a UAV crashes or is lost to enemy fire.

Past success with UAVs in the Persian Gulf War has indicated their usefulness in obtaining timely intelligence during military conflicts and it was during Desert Storm that the first-ever combat tests of Unmanned Aerial Vehicles that were obtained by U.S. forces [Ref. 1]. An example of an effective UAV, the Pioneer remotely piloted vehicle (RPV), has served in numerous fleet and ground operations since 1987 [Ref. 2].

Another advantage of UAVs over manned aircraft is their cost which is only a small fraction of the cost of a manned airplane. Thus they have become an integral part of modern warfare since many missions such as electronic deception, visual identification, laser designation of targets and bomb damage assessment deep in the enemy territory can be performed without endangering any lives or risking any expensive aircraft. They can be launched from practically any type of platform making them ideal for use in the Navy. They are also usually very hard to be detected with radar or infrared systems due to their small size, composite materials and low noise and speed. UAVs are also not limited by the pilot "g" tolerance or fatigue.

B. EVOLUTION OF AN AIRCRAFT

For any type of aircraft the first step in obtaining an accurate mathematical model is to determine stability and control derivatives. These derivatives will impact the flying characteristics and will be used to size control surfaces, design flight control systems and program devices such as simulators.

Three approaches can be taken toward completing this goal. The first and easiest method requires the knowledge of the geometry and inertial properties of the aircraft and employs simple calculations to obtain the derivatives within reasonable accuracy.

The second method involves the use of wind tunnels. However, the results will have to be refined after several scale, interference and dynamic effects are taken into account. This method is much more complicated than the previous one, but usually more accurate.

The final approach which is the most time consuming and costly but with the most promise and precision is flight testing. Dynamic flight test data is used through techniques such as parameter estimation to accurately estimate the stability and control derivatives. Of course limitations in availability of data and noisy measurements can cause serious problems in the successful resolution of all the derivatives of interest.

C. REQUIREMENT FOR MODELING

For the aeronautical controls engineer the goal is to develop a dynamic aircraft model and verify its accuracy. This model will then be used to design a control system for the UAV.

The first step in this process is to develop the high fidelity nonlinear model of the aircraft based upon the predicted flight dynamics involving the stability and control derivatives. This is done as follows:

- The equations of motion are developed and the sum of all forces and moments involved are obtained allowing for an easy conversion into a block diagram representation.

2

- The nonlinear model has to be verified through flight testing and if necessary some computer aided parameter estimation technique should be incorporated to obtain accurate results.

Next the feedback controller is designed through an iterative process employing various methods with the design requirements of response time, overshoot and command and control loop bandwidths being the adjusting knobs to create the desired controller. Flight test is done in three phases:

- On a laboratory stand allowing only a restricted degree of freedom in movement to avoid any damage to the aircraft.

- During a flight with an experienced pilot ready to take over with manual control in the case of any problem.

- Autonomous flight with the aircraft in full operation.

D. STATEMENT OF OBJECTIVE

Over the last several years the Avionics Lab of the Naval Postgraduate School has undertaken the complicated task of developing and implementing control systems for UAVs. Two models have undergone progress up to the point of conducting flight testing. The first one was the Bluebird which was a conventional aircraft and the second one was the Archytas vehicle with a ducted fan propulsion system. The latest vehicle to be used is the Frog unmanned aerial vehicle, a small conventional aircraft acquired as a test bed for designing and testing guidance, navigational and control systems. The objective of this work is to obtain realistic modeling through aerodynamic parameters of the aircraft and through the acquisition of sufficient flight test data to verify the nonlinear model which could be used for purposes of analysis and design guidance, navigation and control systems Ultimately these guidance, navigation and control systems along with a Global Positioning System aided autoland capability will allow for fully autonomous operations.

4

II. AIRCRAFT EQUATIONS OF MOTION DEVELOPMENT

A. NOTATION

First it is necessary to present the notation used throughout this report which is consistent with the previous developments [Refs. 6-8]. Consider Fig. 2 1:

- $\{A\}$ is the coordinate system with basis vectors x_A, y_A, z_A
- ^AP_Q is the position of point Q expressed in $\{A\}$.
- ^AV_Q the velocity of point Q measured and expressed in $\{A\}$.
- $^B(^AV_Q)$ the velocity of point Q, measured in $\{A\}$ and expressed in $\{B\}$.
- ^A_BR is the transformation matrix from $\{B\}$ to $\{A\}$.
- $^A\Omega_B$ is the angular velocity of the $\{B\}$ coordinate system with respect to $\{A\}$ and expressed in $\{A\}$.
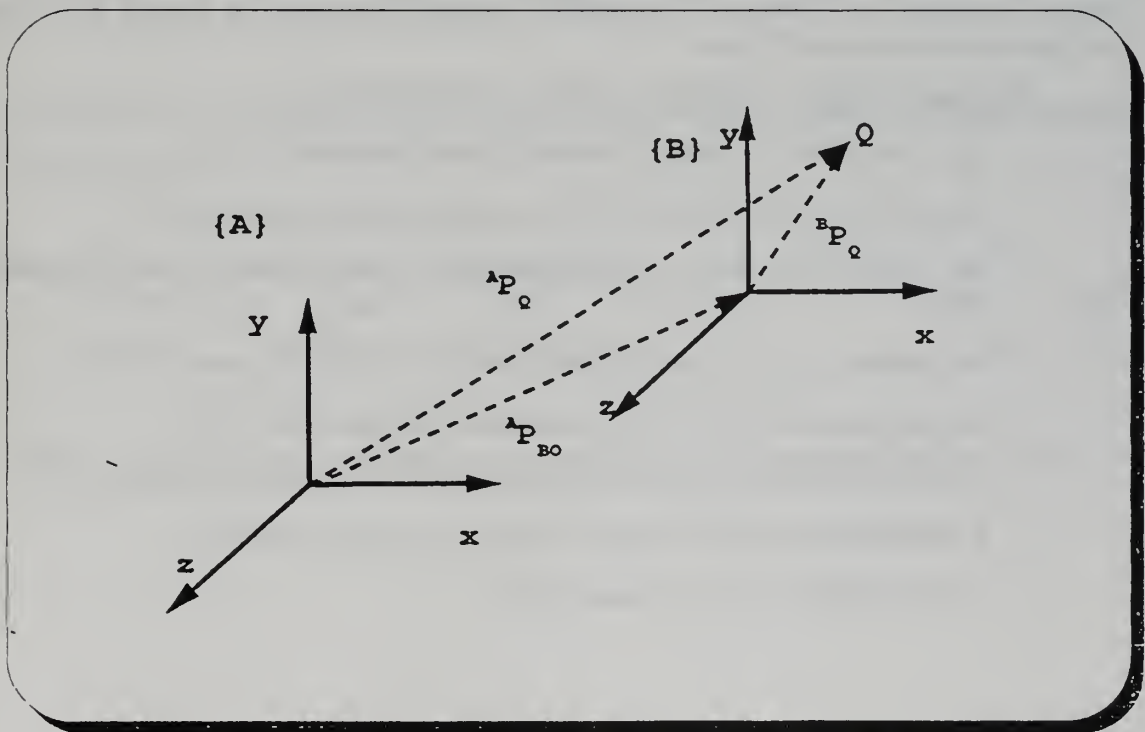- $^B(^A\Omega_B)$ is the angular velocity of $\{B\}$ with respect to $\{A\}$ but now expressed in $\{B\}$.

Figure 2.1 Relative Position of Coordinate Systems

5

B. COORDINATE SYSTEMS

The following coordinate systems and assumptions will be used for the derivation of the equations of motion:

- {U} is the tangent plane coordinate system attached to the Earth.
- {B} is the body fixed coordinate system.
- {W} is the wind axis coordinate system.
- All sensors are located at the cg. (This can be relaxed; however, it is used for simplification).
- The mass of the aircraft remains constant.
- For the vector u its derivative with respect to {B} is $\frac{d}{dt}(u)$ and with respect to. {U} is $(\dot{u})$

The {B} coordinate system is right-handed with X aligned with the thrust axis. All rates p, q, r are defined positive when clockwise looking in the positive axis direction. The {W} coordinate system has X aligned with the wind incident on the aircraft. Thus α is the angle formed by the body x-y plane and the positive X_W axis. The angle β is formed by the body x-z plane and the positive Y_W axis.

For simplification the following definitions are introduced:

- υ_Q is the velocity of point Q, measured and expressed in {U}.
- υ_{BO} is the velocity of origin of {B}, measured and expressed in {U}.
- $\dot{\upsilon}_B$ is the acceleration of {B} with respect to {U}, measured and expressed in {U}.
- $^B\upsilon_Q$ is the velocity of point Q, measured in {U} and expressed in {B}.
- ω_B the angular velocity of {B}, measured and expressed in {U}.
- $^B\omega_B$ is the angular velocity of {B}, measured in {U} and expressed in {B}.
- 0 represents the appropriate size matrix with all elements zero.
- I_n is the identity matrix of dimension n.

6

C. EULER ANGLES

Using the three Euler angles Φ, Θ and Ψ corresponding to roll, pitch and yaw the rotation between any two coordinate systems can be defined. For our case of a conventional aircraft the 3-2-1 order of rotation is used. The singularity point for this order arises at $\Theta = 90^{\circ}$. Thus the transformation from inertial coordinates {U} to body coordinates {B} can be carried out as shown in Fig.2.2. Therefore the direction cosine matrix is expressed in terms of the Euler angles as:

$$
\begin{bmatrix}
\cos\Psi\cos\Theta & \sin\Psi\cos\Theta & -\sin\Theta \\
\cos\Psi\sin\Theta\sin\Phi - \sin\Psi\cos\Phi & \sin\Theta\sin\Phi\sin\Psi + \cos\Psi\cos\Phi & \cos\Theta\sin\Phi \\
\cos\Psi\sin\Theta\cos\Phi + \sin\Psi\sin\Phi & \sin\Theta\cos\Phi\sin\Psi - \cos\Psi\sin\Phi & \cos\Theta\cos\Phi
\end{bmatrix}
\tag{2.1}
$$

Next the kinematic differential equations for the change in Euler angles are given:

$$
\begin{bmatrix} p \\ q \\ r \end{bmatrix} =
\begin{bmatrix}
1 & 0 & -\sin\Theta \\
0 & \cos\Phi & \cos\Theta\sin\Phi \\
0 & -\sin\Phi & \cos\Theta\cos\Phi
\end{bmatrix}
\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix}
\tag{2.2}
$$

The matrix on the right hand side is invertible for all $\Theta \neq \pi/2$ and so the Euler angle rates can be obtained:

$$
\begin{bmatrix} \dot{\Phi} \\ \dot{\Theta} \\ \dot{\Psi} \end{bmatrix} =
\begin{bmatrix}
1 & \sin\Phi\tan\Theta & \cos\Phi\tan\Theta \\
0 & \cos\Phi & -\sin\Phi \\
0 & \sin\Phi\sec\Theta & \cos\Phi\sec\Theta
\end{bmatrix}
\begin{bmatrix} p \\ q \\ r \end{bmatrix}
\tag{2.3}
$$

Therefore the time history of the Euler Angles is obtained integrating equation (2.3).
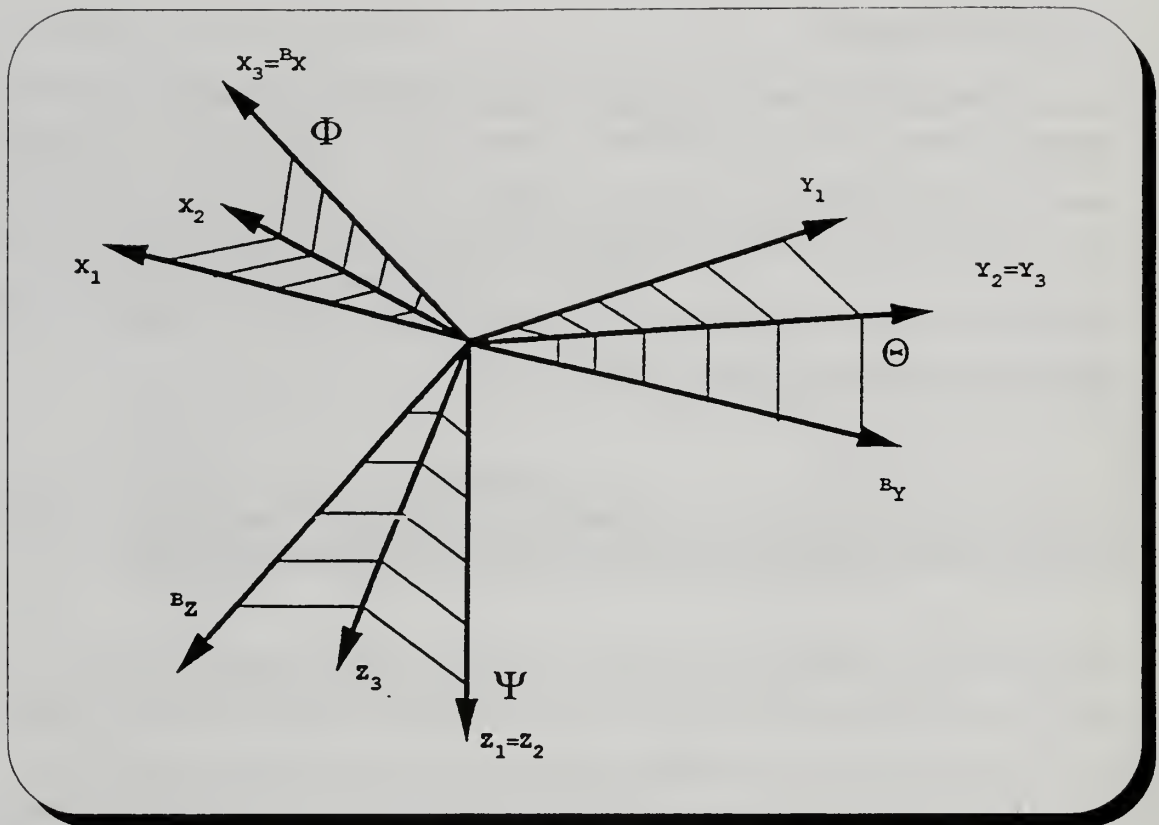
7

Fig.2.2 Z-Y-X Euler Angles Rotation Sequence

D. DERIVATION OF EQUATIONS OF MOTION

For the general body with six degrees of freedom the equations of motion can be derived in two parts. In the first part is the determination of the equations of motion for a rigid body and only the linear and angular momenta is considered. In the second part all the forces are examined and more specifically aerodynamic, gravitational, and thrust forces are taken into account. It is in the modeling of the aerodynamic and propulsive forces where the stability and control derivatives come into picture.

1. Linear Equations

These are derived based upon the Newton's Law, F=ma. However since the measurements are resolved in the body {B} coordinate system the equations are also resolved in the {B} coordinate system. Therefore, using Coriolis theorem we obtain:

$$^BF = m\frac{d^B}{dt}\upsilon_B + m^B\omega_B \times^B \upsilon_B \qquad (2.4)$$

2. Angular Equations

To derive the angular momentum equations Euler's Law of preservation of momentum is employed. Again the equations are written in the {B} body coordinate system for the position of the cg. of the vehicle. Thus we get [Ref. 8]:

$$^BN_{BO} = I_T\,^B\dot{\omega}_B +^B \omega_B \times (I_T\,^B\omega_B) \qquad (2.5)$$

3. State Equations

Having developed the equations of motion, both linear and angular, the next step is to assemble them in a state space form. Thus after rearranging and normalizing we get:

$$\frac{d}{dt}\begin{bmatrix} ^B\upsilon_B \\ ^B\omega_B \end{bmatrix} = \begin{bmatrix} -^B\omega_B \times^B \upsilon_{BO} & + & \frac{^BF}{m} \\ -I_B^{-1}\,^B\omega_B \times (I_B\,^B\omega_B) & + & I_B^{-1}\,^BN \end{bmatrix} \qquad (2.6)$$

4. External Forces And Moments

It is necessary to examine now the external forces and moments acting on the rigid body and distinguish between aerodynamic, propulsive, and gravitational forces:

$$\begin{bmatrix} ^BF \\ ^BN \end{bmatrix} = \begin{bmatrix} ^BF_{grav} +^B F_{prop} +^B F_{aero} \\ ^BN_{prop} +^B N_{aero} \end{bmatrix} \qquad (2.7)$$

9

a. Gravitational Forces

These act on the body and have to be rotated by the appropriate rotation matrix from the tangent to the body coordinate system. This matrix is defined using the Euler angles:

$$^{B}F_{grav} = {}^{B}_{U}R \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} \tag{2.8}$$

b. Propulsive Forces And Moments

These are computed directly in the {B} coordinate system and are:

$$^{B}F_{prop} = \begin{bmatrix} T_x \\ T_Y \\ T_Z \end{bmatrix} \tag{2.9}$$

and:

$$^{B}N_{prop} = \begin{bmatrix} T_l \\ T_m \\ T_n \end{bmatrix} \tag{2.10}$$

where these values depend on the aircraft configuration and are known.

c. Aerodynamic Forces And Moments

The derivation of these forces and moments depends upon the nominal operating point about which they have to be found using a Taylor series expansion. The partial derivatives of the forces and moments with respect to the aerodynamic variables $u/U, \alpha, \beta, p, q, r$ are introduced to get the Taylor series expansion [Ref. 15]:

$$F_{aero} = \delta F_{x'} x' + \delta F_{\dot{x}'} \dot{x}' + \delta F_{\Delta} \Delta + F_0 \tag{2.11}$$

10

Similarly for the moments:

$$N_{aero} = \delta N_{x'} x' + \delta N_{\dot{x}'} \dot{x}' + \delta N_{\Delta} \Delta + N_0 \qquad (2.12)$$

In the above formulae the x' is the state vector:

$$x' = \begin{bmatrix} \frac{u}{U} \\ \beta \\ \alpha \\ \frac{pb}{2U} \\ \frac{qc}{2U} \\ \frac{rb}{2U} \end{bmatrix} \qquad (2.13)$$

and also:

$$\dot{x}' = \begin{bmatrix} \dot{\beta} \\ \dot{\alpha} \end{bmatrix} \qquad (2.14)$$

Finally for the control vector:

$$\Delta = \begin{bmatrix} \delta_e \\ \delta_r \\ \delta_a \end{bmatrix} \qquad (2.15)$$

Combining all the terms together and introducing the matrix of non-dimensional stability derivatives differentiated with respect to. the terms $x', \dot{x}', \Delta$ we get:

$$\frac{\partial C}{\partial x'} = \begin{bmatrix} C_{Lv} & C_{L\beta} & C_{L\alpha} & C_{Lp} & C_{Lq} & C_{Lr} \\ C_{Yv} & C_{Y\beta} & C_{Y\alpha} & C_{Yp} & C_{Yq} & C_{Yr} \\ C_{Dv} & C_{D\beta} & C_{D\alpha} & C_{Dp} & C_{Dq} & C_{Dr} \\ C_{lv} & C_{l\beta} & C_{l\alpha} & C_{lp} & C_{lq} & C_{lr} \\ C_{mv} & C_{m\beta} & C_{m\alpha} & C_{mp} & C_{mq} & C_{mr} \\ C_{nv} & C_{n\beta} & C_{n\alpha} & C_{np} & C_{nq} & C_{nr} \end{bmatrix} \qquad (2.16)$$

11

The term $\frac{dC}{dx'}$ is similar to the previous term; however, only the terms with respect to $\dot{\alpha}, \dot{\beta}$ are computed leaving a 6x2 matrix. Also the matrix of the derivatives with respect to the control inputs elevator, rudder and ailerons is:

$$\frac{dC}{d\Delta} = \begin{bmatrix} C_{L\delta e} & C_{L\delta r} & C_{L\delta a} \\ C_{Y\delta e} & C_{Y\delta r} & C_{Y\delta a} \\ C_{D\delta e} & C_{D\delta r} & C_{D\delta a} \\ C_{l\delta e} & C_{l\delta r} & C_{l\delta a} \\ C_{m\delta e} & C_{m\delta r} & C_{m\delta a} \\ C_{n\delta e} & C_{n\delta r} & C_{n\delta a} \end{bmatrix} \qquad (2.17)$$

Finally the matrix of the constant coefficients is:

$$C_{Fo} = \begin{bmatrix} C_{Do} \\ C_{Yo} \\ C_{Lo} \\ C_{lo} \\ C_{mo} \\ C_{no} \end{bmatrix} \qquad (2.18)$$

The above values refer to the values of the coefficients at the trimming point and not to the values at $\alpha = 0^o$ [Ref. 10]. Also the stability and control derivatives are found in the wind axis coordinate system and it is necessary to transform them from {W} to {B}. The rotation matrix is given below:

$$^B_W R = \begin{bmatrix} \cos\alpha \ \cos\beta & -\cos\alpha \ \sin\beta & -\sin\alpha \\ \sin\beta & \cos\beta & 0 \\ \sin\alpha \ \cos\beta & -\sin\alpha \ \sin\beta & \cos\alpha \end{bmatrix} \qquad (2.19)$$

The aerodynamic forces and moments are premultiplied by this matrix. In addition in order to be consistent with the way lift and drag are defined as positive we have:

12

$$F_{aero} = \begin{bmatrix} -D \\ Y \\ -L \end{bmatrix} \text{ and } N_{aero} = \begin{bmatrix} l \\ m \\ n \end{bmatrix} \qquad (2.20)$$

Finally the most commonly used vector:

$$x = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} \qquad (2.21)$$

is introduced and the complete final expressions for the aerodynamic forces and moments are given:

$$\begin{bmatrix} {}^{B}F_{aero} \\ {}^{B}N_{aero} \end{bmatrix} = \bar{q}\,\bar{S} \begin{bmatrix} {}^{B}_{W}R & 0 \\ 0 & {}^{B}_{W}R \end{bmatrix} \{C_{F_0} + \partial C/\partial x'\, M'x + \partial C/\partial \dot{x}'\, M'\dot{x} + \partial C/\partial \Delta\, \Delta\} \quad (2.22)$$

where M and $\dot{M}$ map from x to x' and from $\dot{x}$ to $\dot{x}'$ respectively. The above expression can now be substituted into the general equation 2.6.

5. Complete Equations Of Motion

All equations presented above 2.8, 2.9, 2.10, 2.22, have to be substituted in the general equation 2.6 to get the state space form. After introducing the terms:

$$ {}^{B}_{W}T = \begin{bmatrix} {}^{B}_{W}R & 0 \\ 0 & {}^{B}_{W}R \end{bmatrix} \text{ and } M_I = \begin{bmatrix} m & 0 \\ 0 & {}^{B}I_B \end{bmatrix} $$

we get the complete set of equations of motion which in the state space form are the following [Ref. 8]:

13

$$\frac{d}{dt}\begin{bmatrix} {}^B\upsilon_{BO} \\ {}^B\omega_B \end{bmatrix} = \chi^{-1} \left\{ \left[\begin{matrix} -{}^B\omega_B\times & 0 \\ 0 & -{}^BI_B^{-1}({}^B\omega_B \times ({}^BI_B^B\omega_B + I_R^B\omega_R)) \end{matrix} \right] + \right.$$

$$M_I^{-1}{}_W^BT\,\bar{q}\,\bar{S}\frac{\partial C_F}{\partial x'}M' \right]\begin{bmatrix} {}^B\upsilon_{BO} \\ {}^B\omega_B \end{bmatrix} + M_I^{-1}\left\{ \begin{bmatrix} {}^BF_{grav} \\ 0 \end{bmatrix} + \right.$$

$$\begin{bmatrix} {}^BF_{prop} \\ {}^BN_{prop} \end{bmatrix}\delta_T + {}_W^BT\,\bar{q}\,\bar{S}\,(C_{F_o} + \frac{\partial C_F}{\partial \Delta}\Delta) \left\} \right] \tag{2.23}$$

$$^U\dot{P}_{BO} = {}_B^U R\,^B\upsilon_{BO} \tag{2.24}$$

$$\dot{\Lambda} = S(\Lambda)^B\omega_B \tag{2.25}$$

$$\chi = I_6 - M_I^{-1}\,{}_W^BT\,\bar{q}\,\bar{S}\,\frac{\partial C_F}{\partial x'}\,M'. \tag{2.26}$$

where **P** is the position vector of the aircraft, and $S(\Lambda)$ is the matrix of kinematic differential equations based upon the Euler angles.

III. THE AIRCRAFT

A. GENERAL DESCRIPTION

The Frog unmanned aerial vehicle, shown in Figure 3.1, is a high-wing tricycle-gear radio-controlled airplane. It is constructed of wood, foam, composites and metal. It is powered by a two-stroke, air-cooled engine with a shaft horsepower of 5.6 hp. It is controlled by a Futaba flight control transmitter operating on a frequency of 72 MHz. To enhance reliability, a factory variant of the control system is available that provides direct control from a transmitter that transmits on 407.275 MHz and this configuration could be used in the event that the primary relay should fail when the aircraft is over the horizon. Table 3.1 describes the physical characteristics of the Frog.

Length	8.125 ft
Height	1.75 ft
Wing Airfoil (est.)	NACA 2415
Horizontal Stab. Airfoil (est.)	NACA 0010-34
$S_{Wing}(S_{ref})$	17.5 ft^2
S_t	3.175 ft^2
S_V	0.9818 ft^2
c	1.66 ft
c_t	0.968 ft
b	10.58 ft
b_t	3.313 ft

b_V	1.25 ft
AR	6.37
AR_t	3.46
AR_V	1.59
V_H	0.485
V_V	0.0235
V'	0.0022
l_t	4.44 ft
l_V	4.44 ft

Table 3.1. Specifications

B. STABILITY DERIVATIVES

The flight condition for which the aircraft model was obtained is described in Table 3.2. Based upon these values the initial estimates of the stability derivatives were made using the physical characteristics of the aircraft such as airfoil data, geometric measurements, relative positions of aircraft components, mass and weight [Refs.11-14]. Theoretical or quasi-theoretical methods taken from the literature were used for calculating these constants thus forming the basis for the first parts of longitudinal and lateral stability determinations [Ref. 12]. These methods are regarded as the most suitable for light aircraft with a typical configuration.

Weight	67.73 lbs.

I_{XX}	12.52 slug*ft^2
I_{YY}	8.43 slug*ft^2
I_{ZZ}	18.55 slug*ft^2
Velocity	60 mph/88 f/sec
Altitude	800 ft MSL
Air Density	0.002327 slugs/ft^3
Center Of Gravity	34.5 % of m.a.c.
$C_{L_{TRIM}}$	0.4295
A.O.A$_{TRIM}$	5.25 °
Elevator$_{TRIM}$	5.14 °

Table 3.2 Flight Condition/Aircraft Configuration

Matlab programs [Ref. 21] written for the physical and derivative calculations are presented in Appendix A. Next the nondimensional stability and control derivatives are shown in Table 3.3 and dimensional stability and control derivatives estimated are shown in Table 3.4.

C_L............................	0.2866	C_D..........................	0.0614
C_{Ladot}......................	4.3034	C_{Do}......................	0.0499
C_{Ladot}......................	1.3877	C_{Madot}......................	-3.7115
C_{Ma}........................	-0.5565	C_{Dde}......................	0.23
C_{Lq}........................	3.35	C_{Mq}......................	-8.8818
C_{Lde}......................	0.3914	C_{Dde}......................	0.0676
$C_{m\,de}$......................	-1.0469	C_{yB}......................	-0.31
C_{nB}........................	0.0575	C_{nP}......................	-0.0509
C_{yP}........................	0.0000	C_{nP}......................	-0.0537
C_{lP}........................	-0.3702	C_{Dde}......................	0.1151
C_{nr}........................	-0.0669	C_{lr}......................	0.1119
C_{yda}........................	0.0000	C_{nda}......................	-0.0272
C_{Lde}......................	0.0000	C_{ydr}......................	0.0926
C_{ndr}........................	-0.0388	C_{ldr}......................	0.0036
C_{Dq}........................	0.0000	C_{yq}......................	0

Table 3.3 Nondimensional derivatives

X_u	-0.1045 /sec	X_a	14.9485 ft/sec
X_{de}	-5.0602 ft/sec^2	Z_{de}	-0.7312 /sec
Z_a	-326.9292 ft/sec^2	Z_{adot}	0.9804 ft/sec
Z_q	2.3667 ft/sec	Z_{de}	-29.3185 ft/sec^2
M_u	0.0000 /ft*sec	M_a	-17.2801 /sec^2
M_{adot}	-1.0869 /sec	M_q	-2.6010 /sec
M_{de}	-32.5049 /sec^2	Y_B	-23.2196 ft/sec^2
$Y_{\dot{a}a}$	0.0000 ft/sec^2	Y_r	0.5181 ft/sec
Y_{da}	0.0000 ft/sec^2	Y_{dr}	-6.9323 ft/sec^2
L_B	-6.7831 /sec^2	L_P	-2.9653 /sec
L_r	0.8964 /sec	L_{da}	24.1120 /sec^2
L_{dr}	0.4849 /sec^2	N_B	5.1744 /sec^2
N_P	-0.2903 /sec	N_r	-0.3619 /sec
N_{da}	-2.4467 /sec^2	N_{dr}	-3.4927 /sec^2

Table 3.4 Dimensional derivatives
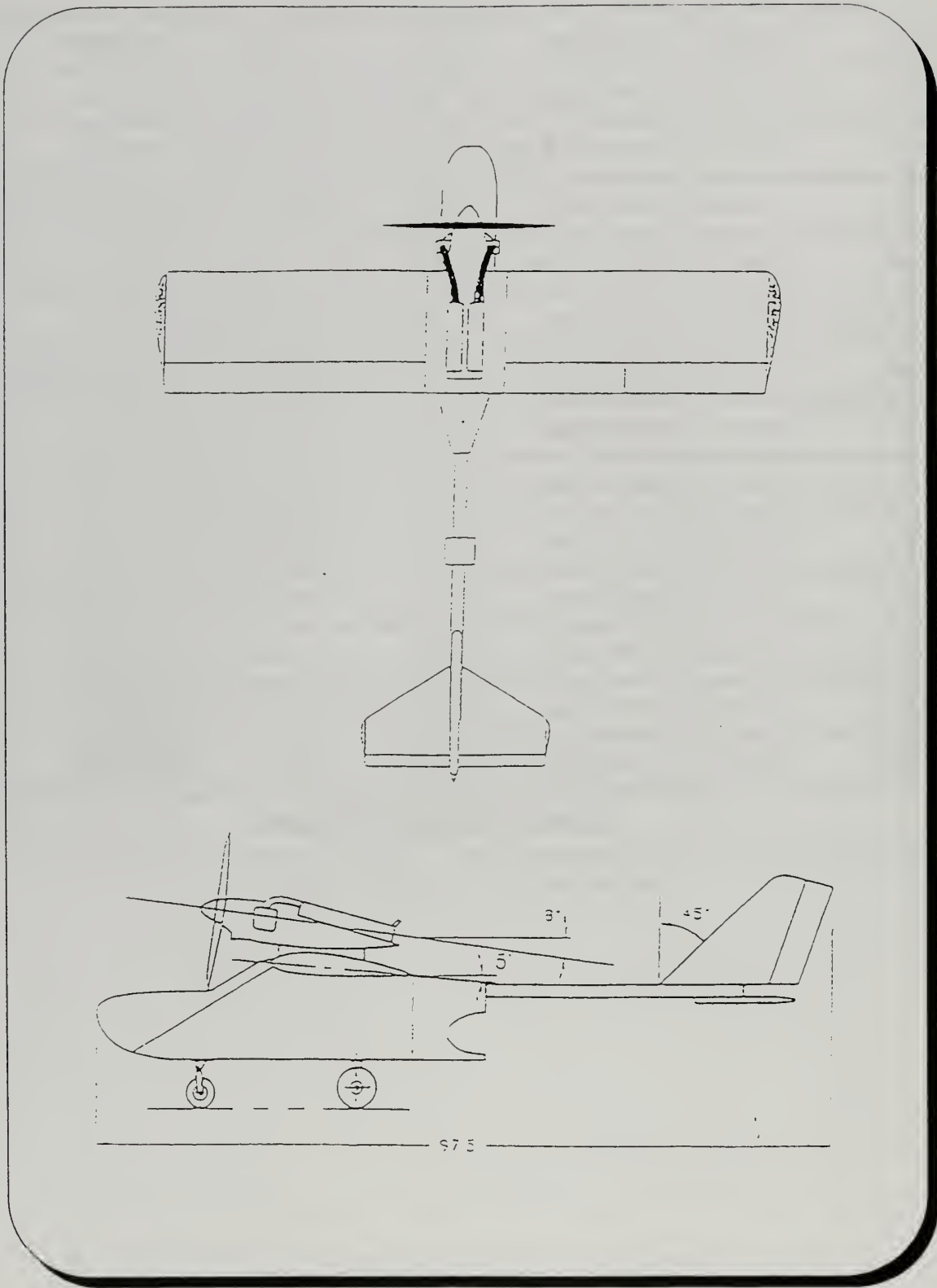
Fig. 3.1 Frog Drawing

20

C. MOMENTS OF INERTIA

When developing a mathematical model of the UAV it is critical to have accurate moments of inertia. Direct calculation of a model's moments of inertia by consideration of the contributions made by individual parts is rather impractical and inaccurate because the model's parts are too small and light and the distances too short to yield anywhere near accurate values for the moments of inertia [Ref. 5]. That is why testing is employed to determine the moments of inertia more precisely in practice. Any changes that might occur in a model's moment of inertia due to addition or substraction of equipment or structure could be calculated directly thereafter.

Moment of inertia is a measure of a rotating body's resistance to acceleration and can be computed by taking the product of the mass of each part of the aircraft and the square of the distance from the aircraft's center of gravity. There are two approaches to obtaining the moments experimentally, each employing the principle of the compound pendulum and each giving the same answer. One method involves hanging the model from a single point in the ceiling on two wires or cords, one well forward of the center of gravity and the other well aft. Thus a compound pendulum is obtained. Another technique is to make the center of gravity of the model itself the pivot point and complete the compound pendulum by hanging a bob weight below it on a pair of fairly stiff wires or lightweight (wooden) struts. Needless to say that the friction at the pivot point should be held to a minimum. Next by giving a small, gentle push in the appropriate direction and timing its oscillations, the oscillatory period (T) is determined which is simply the total number of seconds divided by the total number of cycles (one cycle is one complete swing, to and fro). It should be mentioned that the greater the number of cycles (at least 20-30) and the longer the suspension the greater the timing accuracy, which is vital.

Using the period exhibited and the principles described above the moments of inertia were calculated [Refs. 5, 22]. In order to calculate the moments about all three axes, the model was hung and swung three different ways, each time about the axis of

interest. It was hung by chain and swung as pictured in Figures 3.2, 3.3, 3.4. Specifications for the geometry of each test can be found in Appendix B .

Reference 23 provides equation 3.1 for calculating the moment of inertia for a swinging model:

$$I = \frac{W_{M+S}\bar{Z}_{M+S}}{4\pi^2}p_{M+S}^2 - \frac{W_S\bar{Z}_S}{4\pi^2} - \frac{W_M\bar{Z}_M^2}{g}$$ (3.1)

where W is the weight, $\bar{Z}$ is the distance from the pivot point to the center of gravity, p is the period, and g the gravitational constant. M, S are subscripts referring either to the model or the support.

It was determined that swinging just the support (chains), in the configuration it would be in when supporting the model would not be possible, since the chains would not maintain their positions without the model in place. Equation (3.1) was modified in order to treat the chains as long slender rods and to calculate their moments of inertia as such [Ref. 22]. The new form of equation is:

$$I = \frac{W_{M+S}\bar{Z}_{M+S}}{4\pi^2}p_{M+S}^2 - \frac{W_M\bar{Z}_M^2}{g} - \sum \frac{(W_S)_i(L_S)_i^2}{3g}$$ (3.2)

where L_S is the length of the chain and the summation is taken over all chains (four in this case). In particular, two long and two short chains, all with a weight per unit length ω. After all the appropriate substitutions were made:

$$I = \frac{[W_M + 2\omega\,(L_{SHORT} + L_{LONG})]\bar{Z}_{M+S}}{4\pi^2}p_{M+S}^2 - \frac{W_M\bar{Z}_M^2}{g} - \frac{2\omega}{3g}(L_{SHORT}^3 + L_{LONG}^3)$$ (3.3)

Having the equation in this form all variables are fixed except $\bar{Z}_{M+S}$, $\bar{Z}_M$, p_{M+S}. Therefore for each configuration these three variables were measured and all necessary calculations were made. Three periods were timed during the swing tests and the tests were repeated ten times. The moments of inertia calculated are shown in Table 3.2.
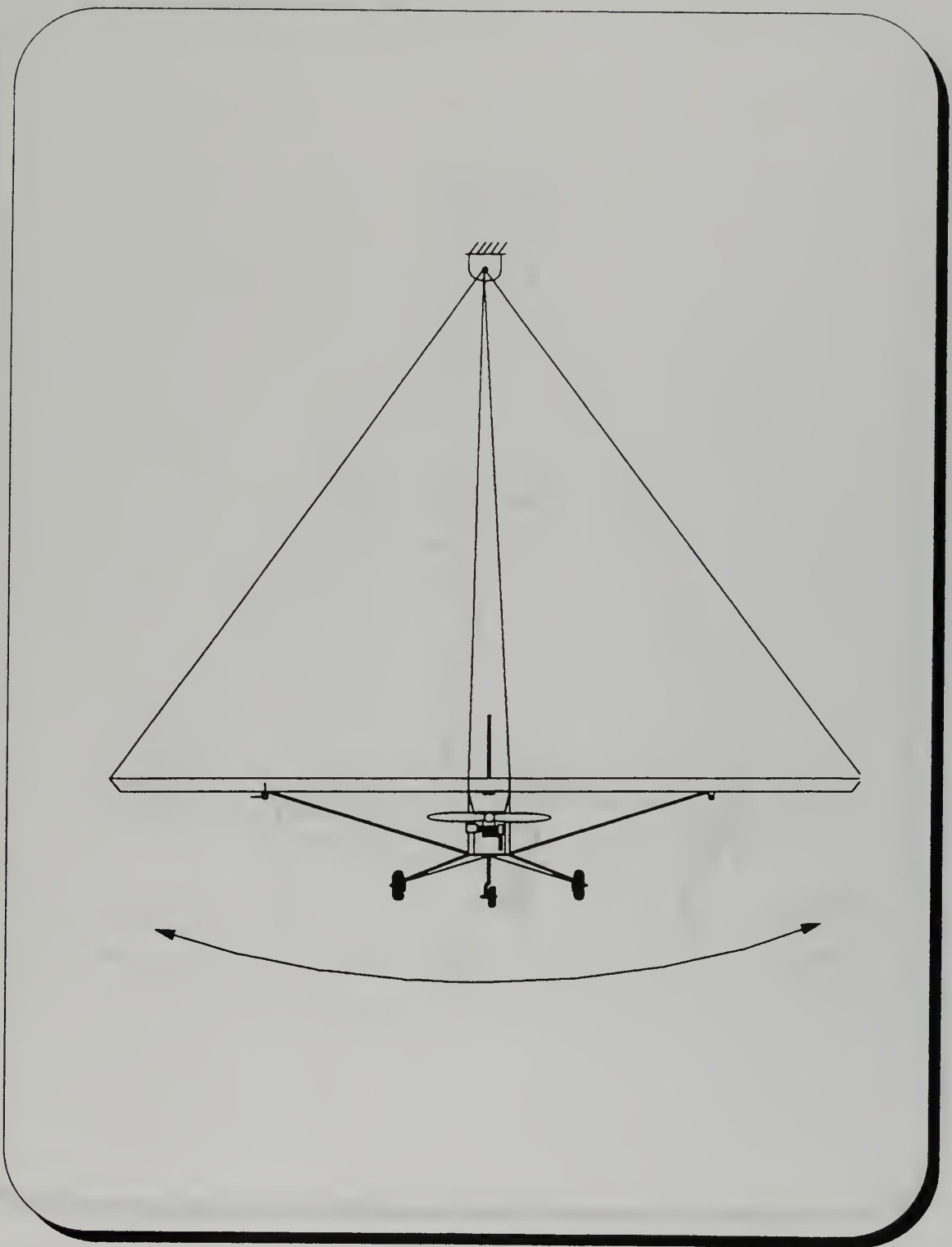
Fig. 3.2 I_{XX} Test
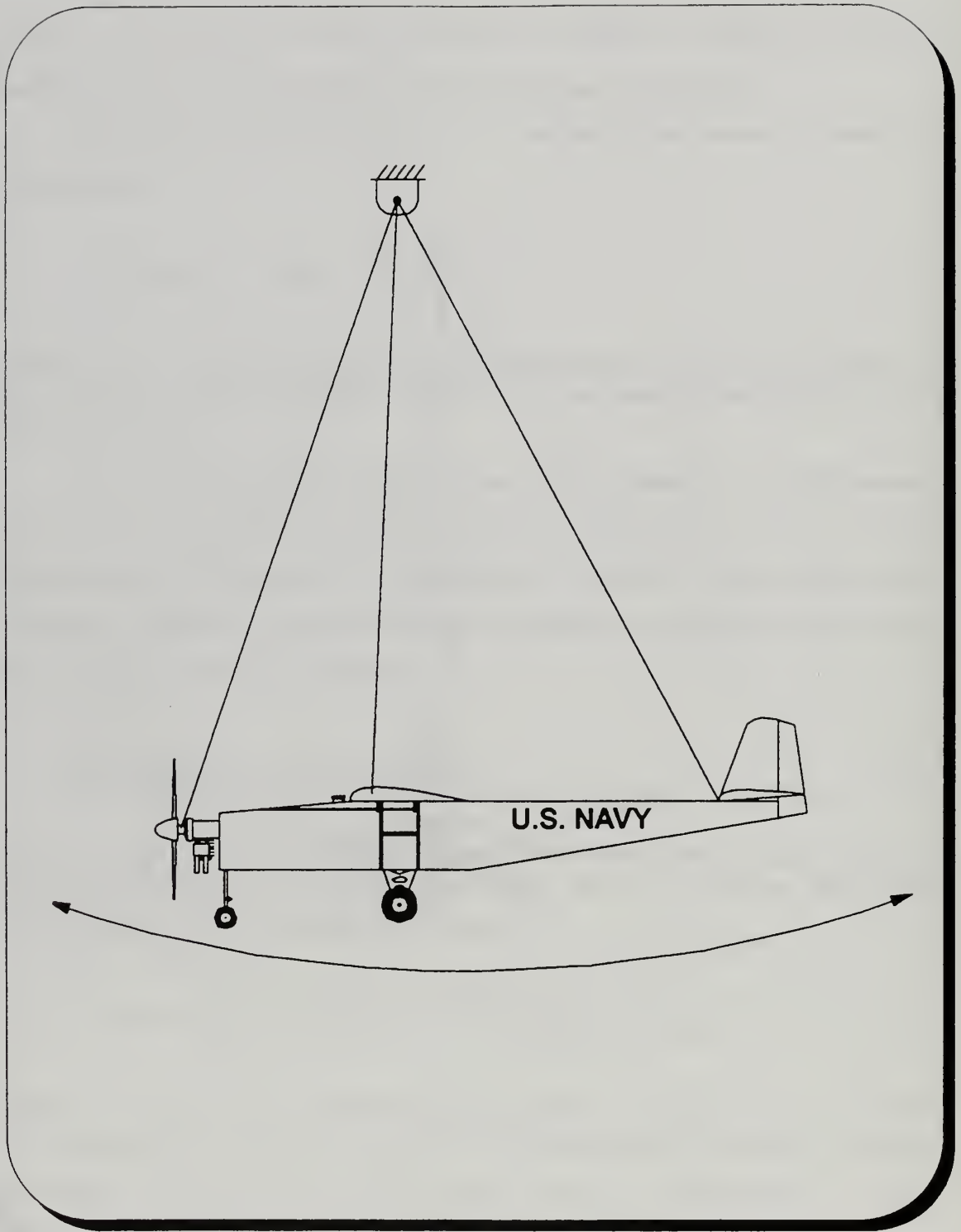
Fig. 3.3 I_{YY} Test
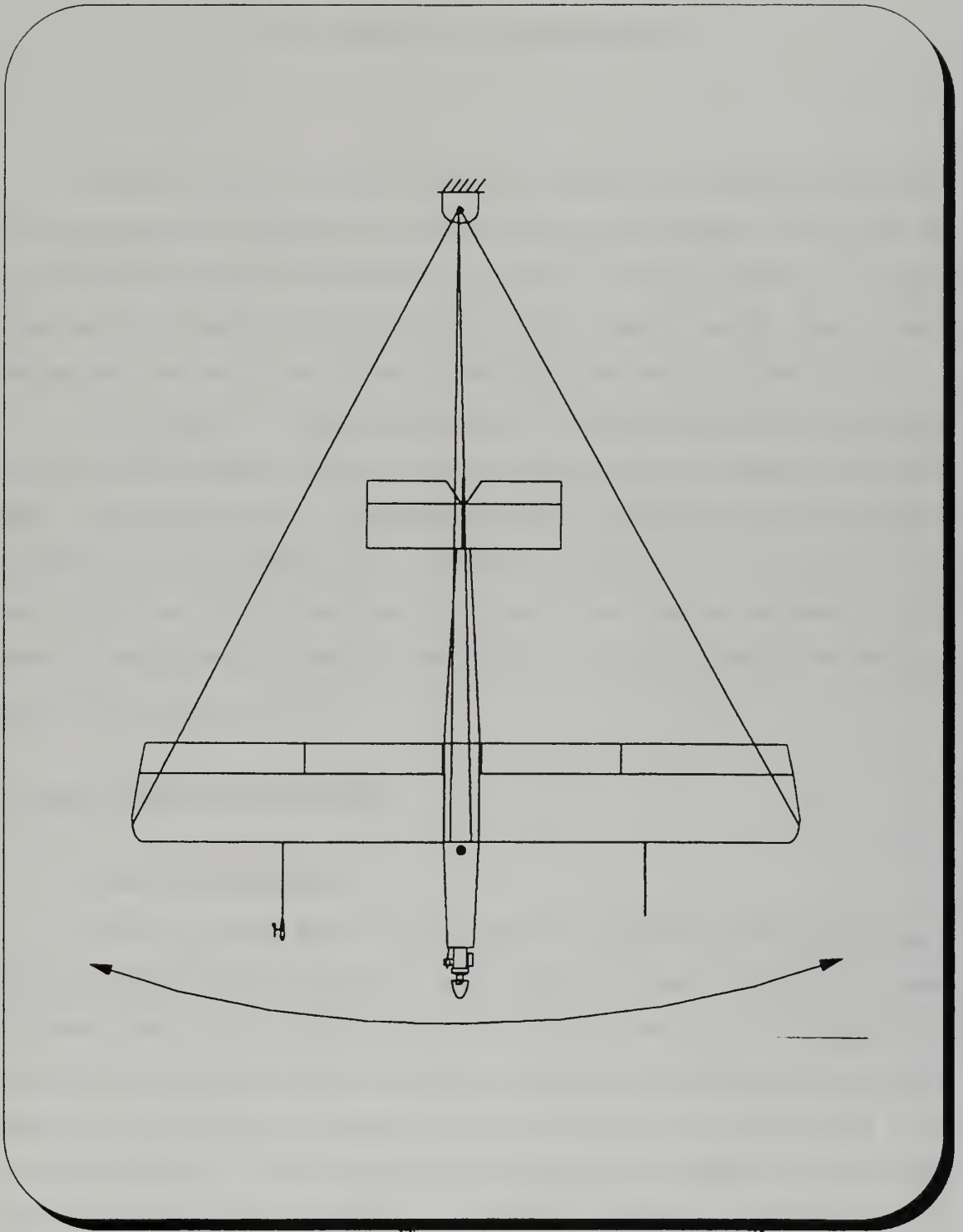
Fig. 3.4 I_{zz} Test

IV. COMPUTER MODELING

In Chapter II the full nonlinear equations of motion were developed and in Chapter III the complete set of stability and control derivatives were obtained. Now the next step is to develop the model of the aircraft on a computer. This model should be in a generic format and should accept values for derivatives from a generic input file. Thus by simply changing the input data for any aircraft the model could be used for a different aircraft.

For this report two already existing models was employed, the first one developed by D.R. Kuechenmeister implementing the equations of motion in Matlab and Simulink [Ref. 6] and the second one in Systembuild [Ref. 19]. These models have been validated by inputting the appropriate data for a well known aircraft, such as the Cessna 172 and comparing the results of the model to existing data. It was found that the results from the numerical linearization are quite close and therefore the equations used in the model are considered to be correct [Ref.6].

A. BASIC NONLINEAR MODEL

1. Basic Simulink Model

The basic nonlinear model was constructed by implementing the state derivative equations in SIMULINK. This model is shown in Figures 4.1 and 4.2 below. In this model the Matlab function blocks abmat.m and frog_data.m are used to input the equations of motion and the stability and control derivatives, respectively. Notice that the force due to thrust since no previous measurements existed, was assumed to be equal to drag. It was decided to follow this simplified approach at this stage since no sufficient information was given about the engine in terms of torque or propeller parameters. In the previous works the term of thrust was computed from an assumed propeller efficiency and the given horsepower of the engine [Refs. 6, 7]. However it has been found [Ref. 4] that especially

27

the value of horsepower of the engine given from the manufacturer is far from accurate and is not attained in flight. A more detailed description of the procedure for determining the shaft power from ground torque tests and wind tunnel tests for propeller characteristics is given in [Ref. 4]. Therefore the propulsive forces were:

$$F_{PROP} = \begin{bmatrix} D \\ 0 \\ 0 \end{bmatrix} \delta T \qquad (4.1)$$

and for the propulsive moments using the position of the propeller from the center of gravity we obtained that:

$$N_{PROP} = {}^B P_{eng} \cdot D \cdot \delta T \qquad (4.2)$$

where ${}^B P_{eng}$ is the position of the engine in {B} body coordinate system. In Appendix C all the input values are found in file Frog_dat.m and the file Abmat.m for the equations of motion.
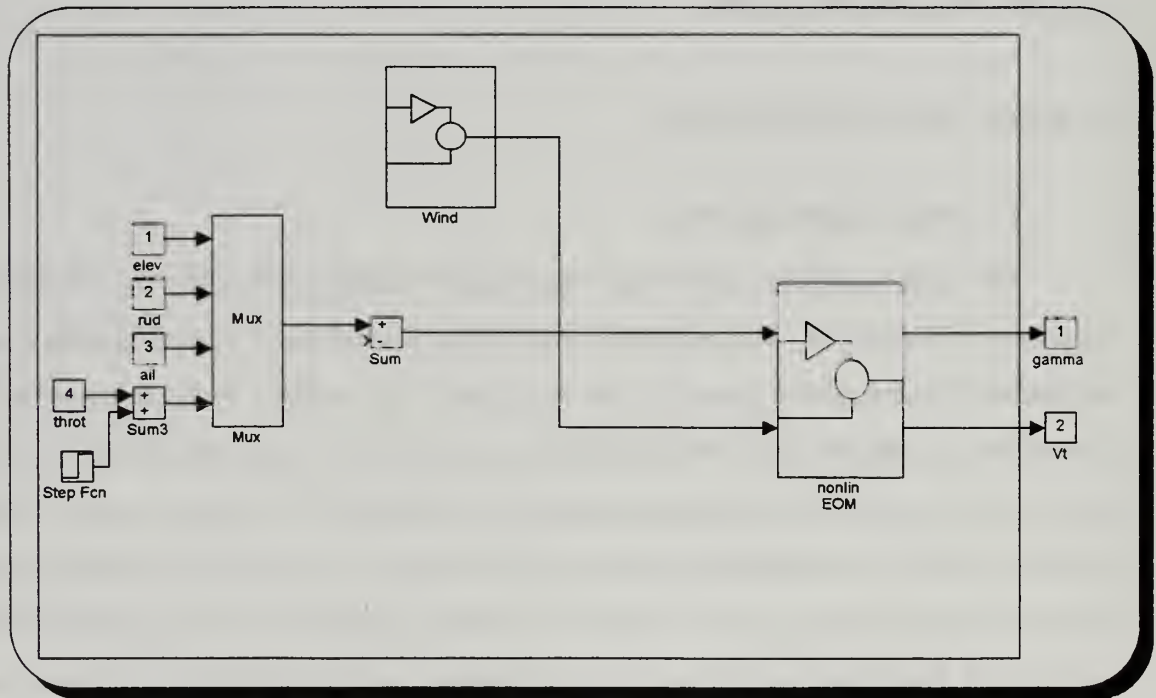
Fig. 4.1 Equations of Motion Implementation

Next the Simulink diagram for the block of nonlinear equations of motion of Figure 4.1 is presented in Figure 4.2. In this model the airspeed and the flight path angle have been selected as outputs since they will be used later in the trimming and linearization of the model.
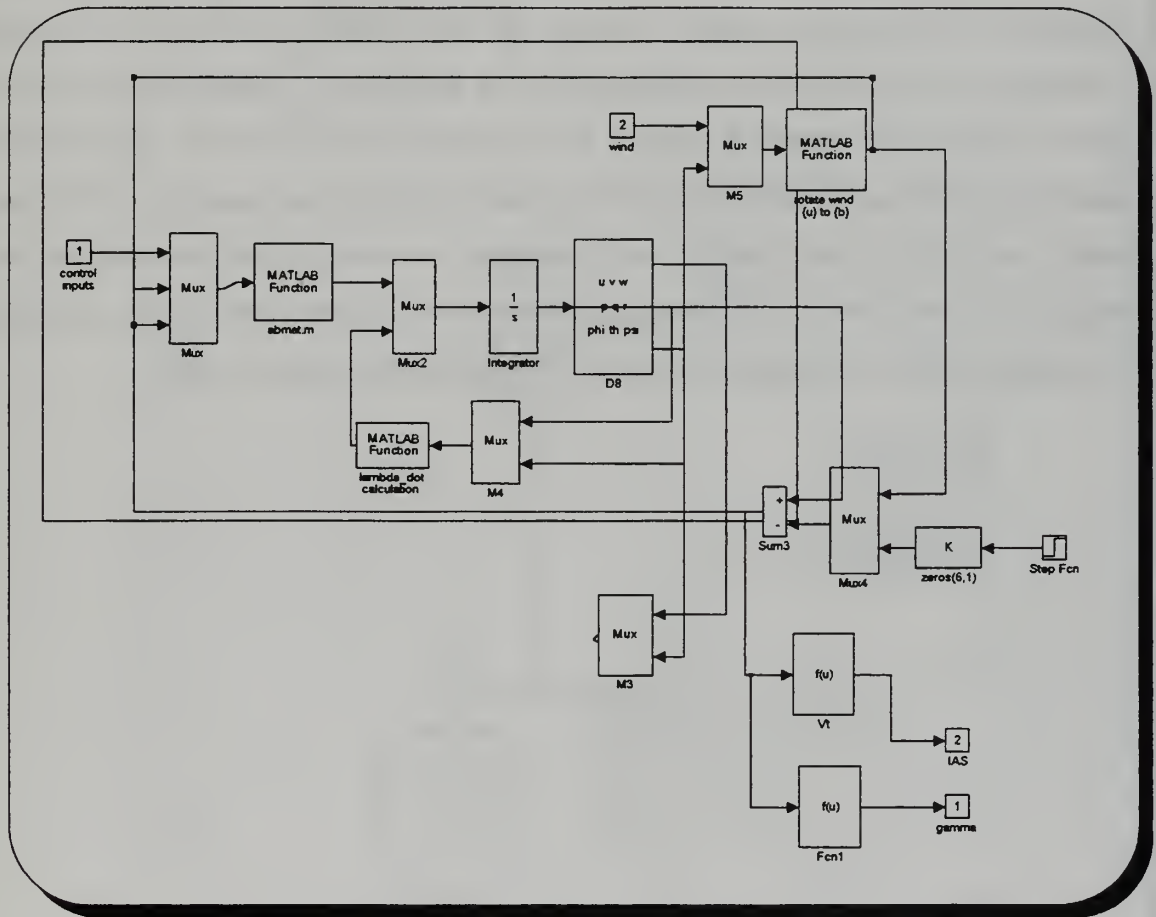
Fig. 4.2 Nonlinear Equations of Motion Block

2. Basic Systembuild Model

The state derivative equations were also implemented on Systembuild in a similar manner. Xmath/Systembuild is a software program similar to the Matlab/Simulink software program developed by Math Works Inc. It is suitable for both the classical input/output control techniques and the modern state-space representations. Specifics can

29

be found in [Ref. 19] and the Xmath and Systembuild Core manuals. With Systembuild using a hierarchical method of organization, based on the Superblock concept any model can be drawn and tested. Therefore the nonlinear equations of motion were implemented as shown in Figures 4.3. All the Superblock diagrams are presented in Appendix E.

In the superblock of the Integrators the derivatives of the state vector are integrated to obtain linear, angular velocity, the Euler angles and position in cartesian coordinates. The dynamics are implemented in the superblock "Dynamics_Euler" and the data for the specific aircraft is input to the superblock of "Aero_forces_and_moments" inside the superblock of "Dynamics_Euler". Thus the differential equations of the linear, angular and Euler angles equations are obtained in the corresponding superblocks which produce the derivatives of the state vector. Moreover the same assumptions concerning the thrust applied by the engine were made as in the Matlab/Simulink model.

Fig. 4.3 Frog Systembuild diagram

31

B. LINEARIZATION OF DEVELOPED MODEL

The linearization process started with the trimming of the equations of motion at the nominal flight condition of $V_T = 88$ *ft*/ sec and $\gamma = 0$. Furthermore it was also selected that for a cruise condition the bank angle should be $\phi = 0$ and sideslip $\beta = 0$. Therefore the complete set of the nine equations of motion were solved for the rest of the state vector and input vector which were unknown in trim using the "trim" command in Matlab:

$$x_0 = \begin{bmatrix} 88 & 0 & 0.1593 & 0 & 0 & 0 & 0 & 0.0018 & 0 \end{bmatrix}'$$

$$u_0 = \begin{bmatrix} -0.0431 & 0 & 0 & 0.9805 \end{bmatrix}'$$

then the output will be (airspeed/sideslip):

$$y_0 = [88; 0]'$$

Next the linearization was obtained in Matlab using the "linmod" command and the matrices A, B, C, D were found along with the eigenvalues of the A matrix and are presented in Table 4.1.

Eigenvalues	Damping	Freq.(rad/sec)
0.0474	-1.0	0.0474
0	-1.0	0
-0.0293+0.5597i	0.0522	0.5604
-0.0293-0.5597i	0.0522	0.5604
-0.1964+2.4972i	0.0784	2.5049
-0.1964-2.4972i	0.0784	2.5049
-3.2691	1.0	3.2691

-3.7591+3.5964i	0.7226	5.2024
-3.7591-3.5964i	0.7226	5.2024

Table 4.1 Eigenvalues of Linearized model

The complete numerical results of the linearization process are presented in Appendix D. It is easy to identify the various modes of the longitudinal and lateral/directional dynamics of the aircraft [Ref. 24]. The two pairs of complex eigenvalues:

$$\lambda_{SP} = -3.7591 \pm 3.5964i$$

and:

$$\lambda_P = -0.0293 \pm 0.5597i$$

correspond to the short period and phugoid modes respectively. It may be noted that the short period is very highly damped and with a natural frequency of $5.2024\ rad/\sec$. It is therefore within the satisfactory boundaries of the short period "thumbprint", although the response initially could be a bit abrupt. The phugoid mode is very lightly damped ($\zeta = 0.0522$) as expected and with a very low natural frequency of $0.5604\ rad/\sec$.

In the lateral/directional dynamics the dutch roll mode can be easily identified with the following pair of complex eigenvalues:

$$\lambda = -0.1964 \pm 2.4972i$$

and has a damping ratio and natural frequency:

$$\zeta_{D-R} = 0.0784; \qquad \omega_n = 2.5049\ rad/\sec$$

The dutch roll damping is characterized as low to moderate and therefore the response is apparent but should not give serious difficulty in maneuvers. The value of the natural

frequency in dutch roll is moderate to high and in some cases the airplane may become oversensitive.

We also observe that the roll response has a stable eigenvalue of:

$$\lambda = -3.2691$$

and therefore has a time constant of :

$$\tau = 1/\lambda = 0.3059 \ sec$$

for time to half amplitude.

Finally the only unstable mode is the spiral with an eigenvalue:

$$\lambda_{SPIRAL} = 0.0474$$

with a time constant of 21.097 sec which is considered as rather low. Due to a rather large derivative of N_β the spiral mode is divergent and may affect the flying qualities of the aircraft since it could result in difficulty in lateral trimming for wings level flight.

V. FLIGHT TESTING

The primary goal of this research was to investigate the flying qualities of the UAV Frog. The nonlinear model which has been developed cannot be used to design any useful and realistic controller unless flight testing verified its accuracy. It is therefore desirable:

- To correlate the analytic parameter estimates with flight test data.
- To more accurately refine the parameter estimates for purposes of control system analysis and design.

It seems wise, therefore, to use flight-test data, at the very least, as a verification tool of aircraft stability and control derivatives. Ultimately a more sophisticated computer-aided parameter estimation routine will be incorporated that will lead to the accurate extraction of the stability and control derivatives.

Flight tests were conducted using the UAV Frog at an outlying field near the Naval Postgraduate School. These tests involved transporting the Sun Sparc 2 workstation Intrepid, the luggable PC AC100, the communication box, the RF antenna, the portable generator, and the airplane to the field. All the appropriate connections and necessary steps are described in [Ref. 19]. Before any flight test could be conducted, all the calibrations of actuators and sensors were ensured. The actual flight test was then commenced and data collection was initiated. The data saved was finally converted to a format suitable to be analyzed in Xmath. One of the benefits of the Matrix$_x$ software was the ability to collect and analyze flight test data during the flight testing process. For a complete description of all steps see [Ref. 19].

A. LONGITUDINAL DYNAMICS

Due to limitations in time and resources one dynamic mode of the longitudinal dynamics was mostly studied. The short period mode of an airplane is the one of the two modes of the longitudinal dynamics and concerns the pitching motion of the plane about

the center of gravity with little or no airspeed variation, characterizing the ability to return to the trim angle of attack following some disturbance. The other longitudinal mode, the long period mode or phugoid is a gradual interchange between potential and kinetic energy. The phugoid is in general slow and controllable by the pilot while the short period is not and therefore subject to tight specifications. Since the short period is the mode affecting most of the longitudinal flying qualities, it was singled out for study.

To evaluate the aircraft performance the instrumentation measured the following critical parameters:

- angle of attack, α
- pitch rate, q
- elevator deflection, δ_e

Also the airspeed could only be obtained through the GPS data recordings since no accurate airspeed measurement unit was installed when the tests took place. It has also been found that the pendulums used to measure the angles give accurate results only during steady state conditions and could not be relied during any kind of maneuver; therefore their measurements are not presented.

The measured and recorded elevator deflection was the input to the nonlinear model of the aircraft in Matrix$_x$ and the outputs consisting of angle of attack and pitch rate were plotted versus the flight test results. Five individual runs were conducted during the flight test, and for each run an elevator doublet was applied [Ref.24]. To excite the short period mode only, the goal was to have as little deviation as possible in pitch attitude from trim at the end of the doublet. The technique used was as follows:

- trim
- apply $5 - 10^o$ nose down, then apply $5 - 10^o$ nose up
- release and record the aircraft's response

In Figure 5.1 the results for the first run are presented. The full results for all runs can be found in Appendix F.
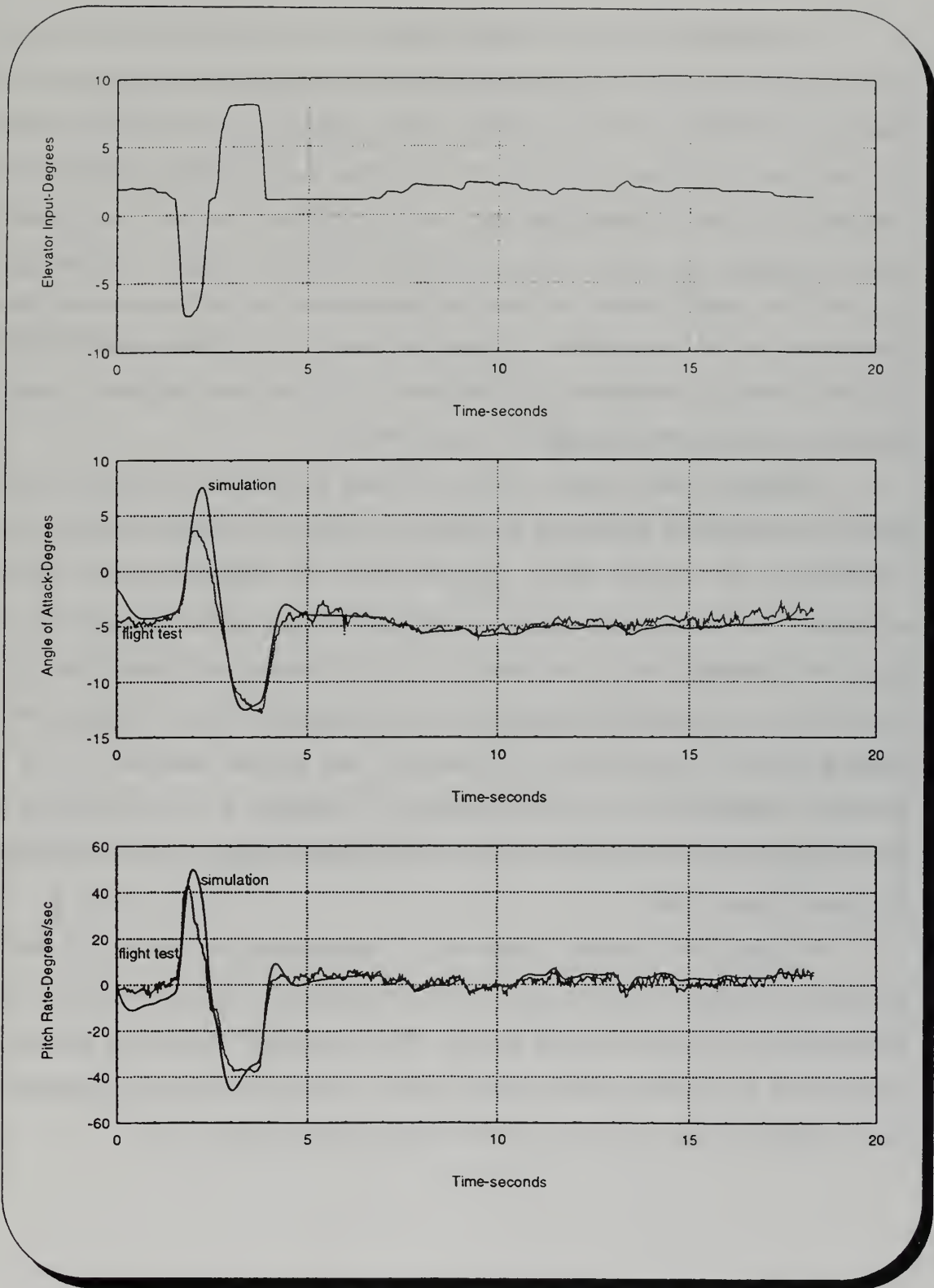
Fig. 5.1 Nonlinear model response plotted over flight test data/ First maneuver

37

It is apparent that there is a close agreement between the flight test data and the response of the nonlinear model. In both cases the short period damping was quite high, resulting in an absence of overshoot while the initial response is very slow since the natural frequency has a moderate value [Ref. 24]. (In some cases the peaks of the aircraft's response in both angle of attack and pitch rate look like they have been cut off and this could be identified as mostly a sensor problem.) As another aside a constant bias of $\alpha_{bias} = 1.5°$ in angle of attack was added in all cases due to a difference in trim values. Nevertheless the close agreement verifies the values of the longitudinal parameters calculated previously analytically. It is therefore, a valid nonlinear longitudinal model for purposes of control system analysis and design [Ref. 25].

Although the short period mode was mostly pursued, for comparison reasons mainly, the response of the aircraft in airspeed and altitude was plotted versus the one obtained from the nonlinear model. The GPS data was employed since no accurate information was obtained for airspeed and altitude from the Inertial Measurement Unit. From the Systembuild model it was easy to extract the airspeed and altitude. Due to the much lower frequency the GPS data exhibits a step behavior. In Figure 5.2 the altitude and airspeed variations are plotted for both the flight test and the simulation for the first maneuver, while the full set of plots are presented in Appendix F. It was found that they are in a relatively close agreement and thus the longitudinal nonlinear model is verified for the phugoid mode as well.

For even more accurate predictions of the longitudinal control and stability derivatives a computer aided parameter estimation program should be incorporated once a full state vector could be measured in flight. This will provide the ability to estimate the parameters in the presence of state noise; however, a more accurate and comprehensive data acquisition system should be employed for this purpose [Refs. 17, 18].
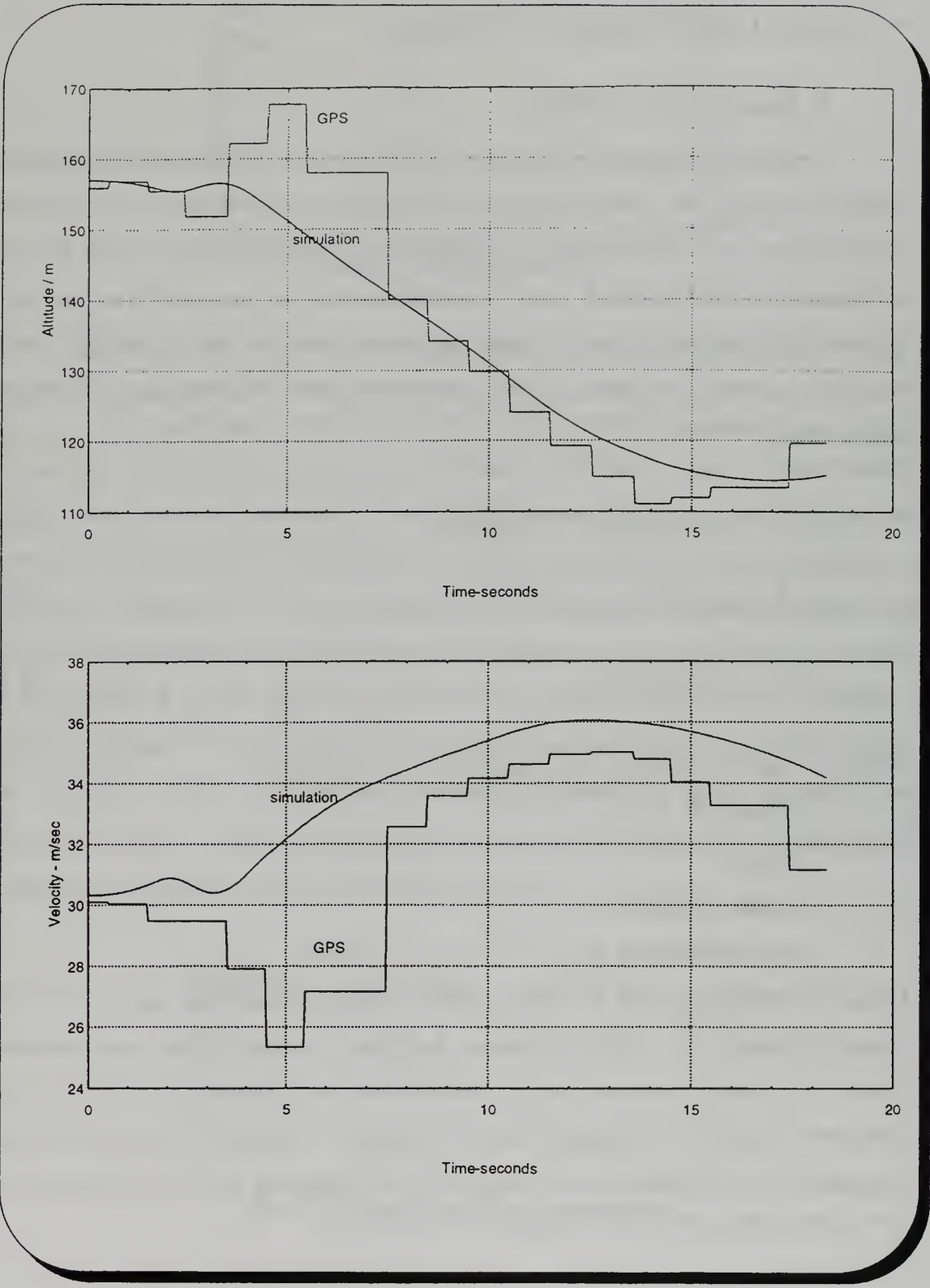
Fig. 5.2 Phugoid response of nonlinear model over flight test data/ First maneuver

39

B. LATERAL DIRECTIONAL DYNAMICS

1. Steady Heading Sideslips

To study the behavior of the aircraft in terms of the lateral/directional dynamics as well as to evaluate the fidelity of the nonlinear computer model a series of steady heading sideslip maneuvers were conducted. By measuring the control deflections and forces that are required to hold the aircraft away from trim the equal and opposite forces that tend to stabilize the plane can be found. During this steady maneuver the yaw and roll rates are both zero and therefore their contributions are also zero. The relations governing this maneuver are:

$$C_{Y_B}\beta + C_{Y_{\delta_r}}\delta_r + C_{Y_{\delta_a}}\delta_a = -C_L \cdot \sin(\phi) \tag{5.1}$$

$$C_{n_\beta}\beta + C_{n_{\delta_r}}\delta_r + C_{n_{\delta_a}} = 0 \tag{5.2}$$

$$C_{l_\beta}\beta + C_{l_{\delta_r}}\delta_r + C_{l_{\delta_a}} = 0 \tag{5.3}$$

During the flight test the following parameters were recorded:
- sideslip, β
- airspeed
- rudder deflection, δ_r
- aileron deflection, δ_a

Using the measured values of rudder, aileron deflections and bank angle for various values of sideslip, the plots of deflection, bank angle versus sideslip were obtained in Figure 5.3. The response was found to be fairly linear as expected and the slopes of these lines were extracted. The same slopes were also computed from the theoretically calculated values of the derivatives using the above equations 5.1-5.3 and found to be in quite close agreement with the exception of the slope ϕ/β:

Ratio	Flight Test	Theoretical
δ_r/β	1.286	1.303
δ_a/β	0.265	0.255
ϕ/β	1.353	0.441

Table 5.1- Sideslip Results

Based upon the results for the slopes from the flight testing the values of the β derivatives were calculated from equations 5.1-5.3. This yielded improved values of the C_{Y_β}, C_{l_β}, C_{n_β} derivatives: -0.70017, -0.05254 and 0.057085 respectively and in Table 5.2 the old and the improved values are presented. With these values the steady heading sideslip maneuvers were simulated in the nonlinear model and the response was compared to the actual aircraft behavior in Figure 5.4 while the full set of these plotted responses can be found in Appendix F. It is easily seen that the simulation is in quite good agreement with the flight test data. Gust effects are not modelled and may provide some discrepancy in the analysis. It is also apparent that the level of electronic noise is quite significant in both the response of the aircraft and the control inputs and to avoid unrealistic inputs to the model the control input data was processed by deleting any spikes caused by noise. However this is a steady state response and in general there is an acceptable match between the nonlinear model and the plane's response.

Derivative	Old value	New value
C_{Y_β}	-0.31	-0.70017
C_{l_β}	-0.0509	-0.05254
C_{n_β}	0.0575	0.057085

Table 5.2- Sideslip derivatives comparison
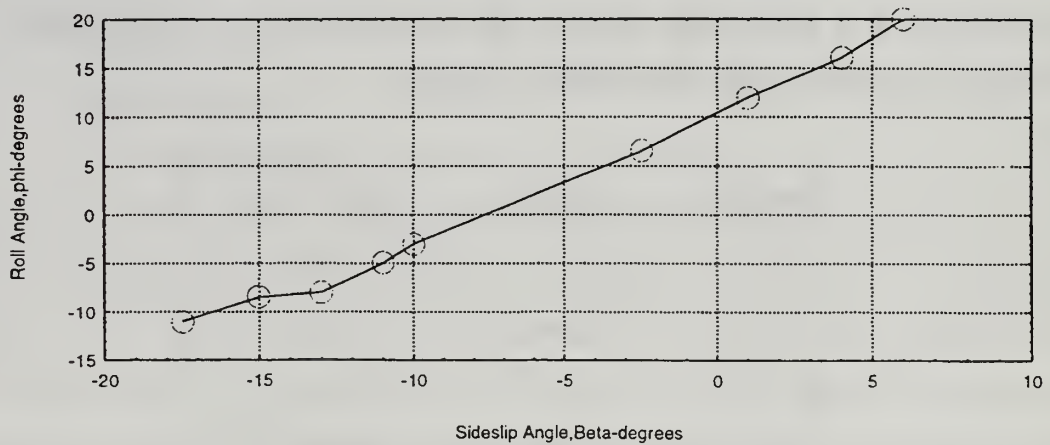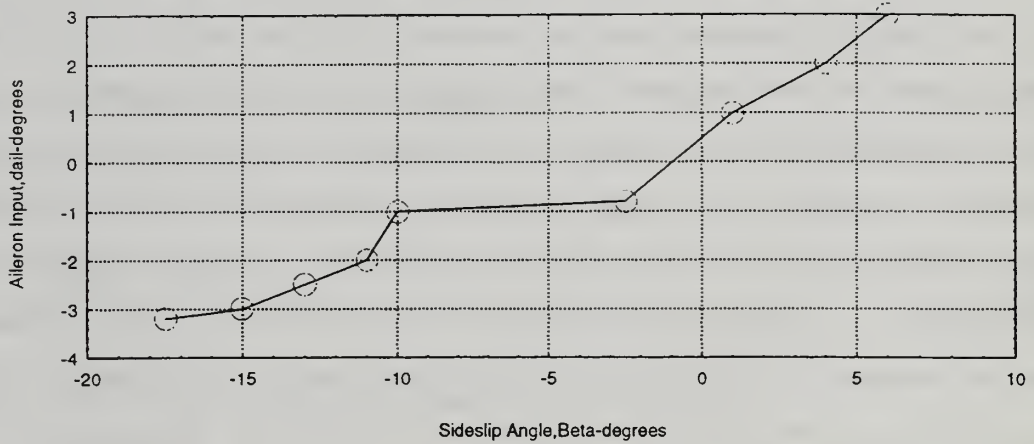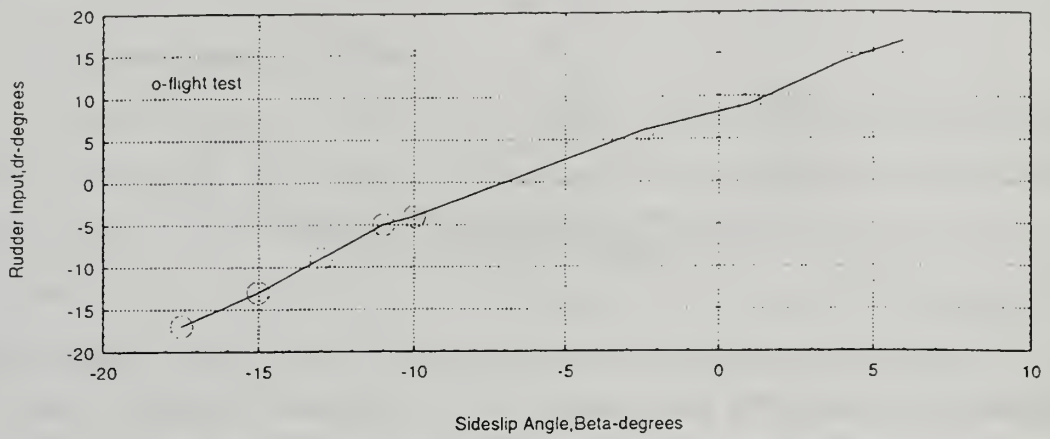
Fig. 5.3 Rudder, Aileron, Bank angle vs. Sideslip Angle, trimmed sideslips

42
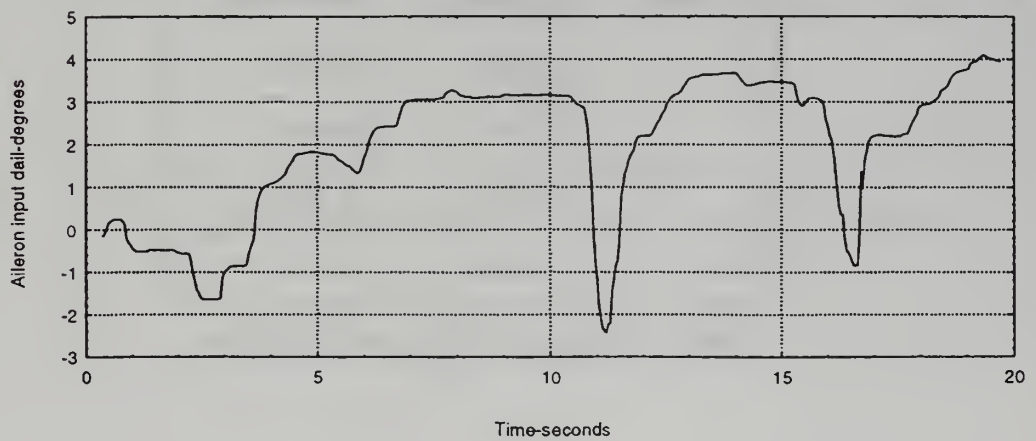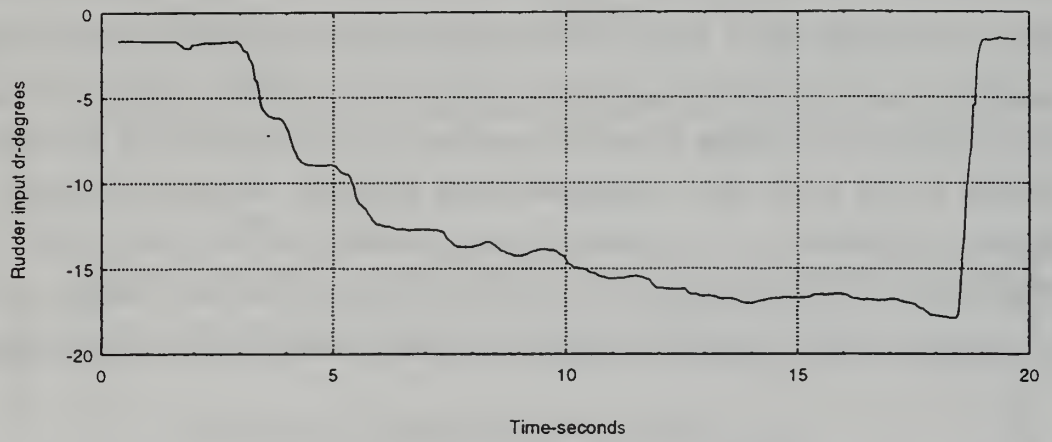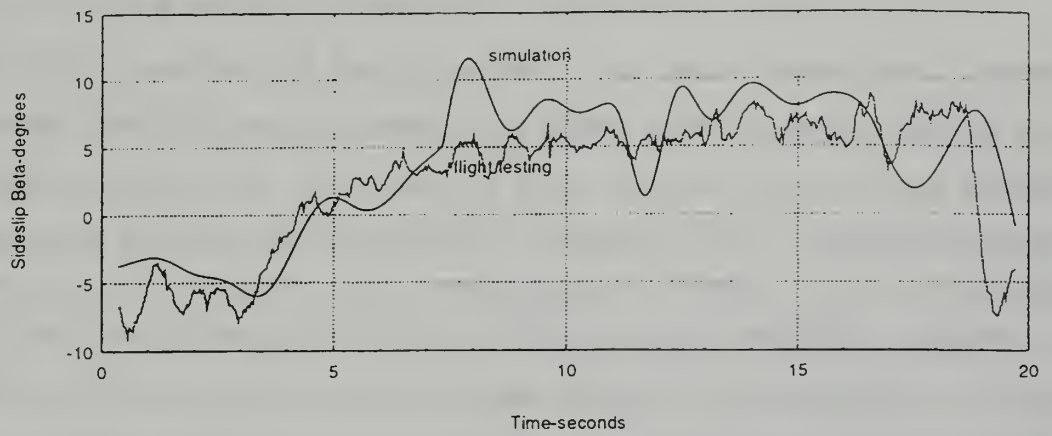
Fig. 5.4 Steady Sideslip Response of Computer Model over Flight Test Data

43

2. Dutch Roll/Spiral Response

To further investigate the fidelity of the computer model the dutch roll mode was excited in flight testing by applying a rudder doublet and after the plane settled for some time applying an aileron doublet, leading to the spiral response. The same response was obtained from the model and the results in terms of the roll rate and yaw rate are presented in Figure 5.5 and in Appendix F. It was found that adjustment of the damping derivatives: C_{l_p}, C_{l_r}, C_{n_p} and C_{n_r} was necessary to get a relatively close match between the responses. Initially the computer model was much less damped than the actual aircraft and with a different period. By adjusting these derivatives through an iterative process the behavior of the computer model was improved; however, further processing is required to achieve a better agreement. The response in terms of the bank angle is not shown since the pendulums used in the inertial navigation unit are not trustworthy. Another problem was that no channel was available during this maneuver to record the sideslip β since this is not obtained as one of the IMU parameters. After adjusting, the improved values of the damping derivatives C_{l_p}, C_{l_r}, C_{n_p} and C_{n_r} are: -0.3702, 0.4476, -0.1074 and -0.1338 respectively. It was found that it was necessary to decrease them (more negative) in order to make the nonlinear model more damped and match it with the actual plane's response.

Derivatives	Old values	New values
C_{l_p}	-0.3702	-0.3702
C_{l_r}	0.1119	0.4476
C_{n_p}	-0.0537	-0.1074
C_{n_r}	-0.0669	-0.1338

Table 5.3 Damping Derivatives Comparison
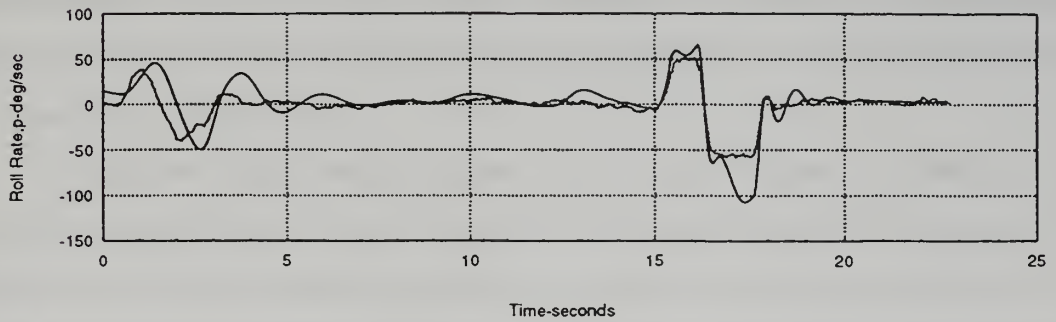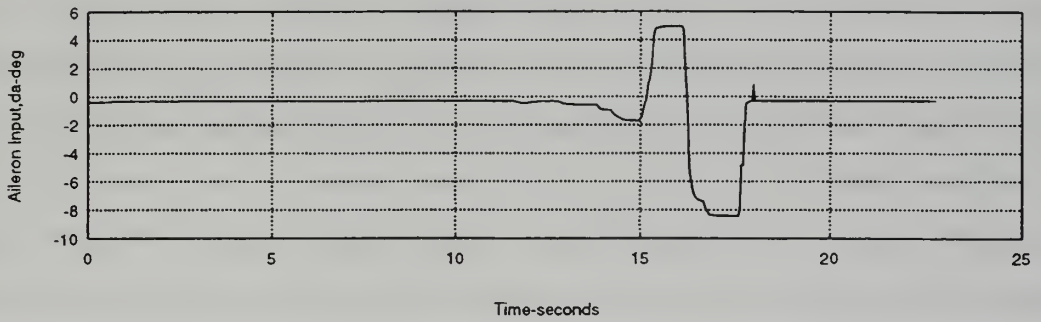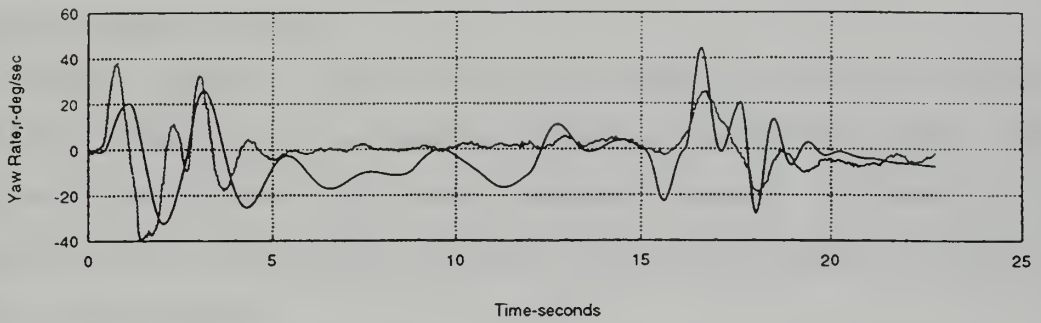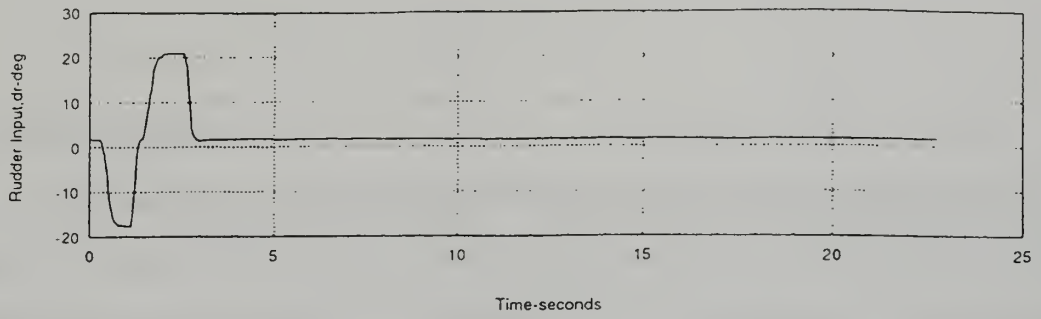
Fig. 5.5 Angle Rates Response in Rudder/Aileron Doublet

45

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

A determination of the moments of inertia through the compound pendulum analysis was done with reasonable results. Future changes to the configuration of the aircraft could be accounted for by adjusting for the contribution of each individual component added or removed.

Initial estimates of stability and control derivatives were made by means of conventional aircraft-design-type methods. Taking into account characteristics such as lift-curve slopes, geometry and weight parameters measured the derivatives were estimated in Matlab with programs which allow for any future changes or for considering different flight conditions.

A high fidelity nonlinear model of the Frog was implemented in $Matrix_x$ /Systembuild using the superblock diagram structure. This structure allows for changes to be easily made and requires little or no programming ability, other than some familiarity with this software. One significant benefit of this product is the ability to collect and analyze data even during the flight testing process. This allowed us to make changes and record different data in the field without having to dismantle the equipment. The real-time data acquisition allowed to convert the data of the flight test to a form suitable for analysis purposes. Since all inputs and outputs could be recorded at each time step thus the data was scrutinized after each test.

Longitudinal flight tests supported with a great success the fidelity of the parameters estimated both in short period and in phugoid modes. The results of the simulation were almost identical to the flight testing data and were both dictated by the strong damping and the limited response. In the lateral-directional tests the comparison between the simulation and the flight test can be described as quite promising; the response is also very stable with the exception of the spiral which however does not

represent a serious problem since it has a time to double of about 20 seconds. However, the pilot should be aware of this tendency of the aircraft to avoid trim problems.

B. RECOMMENDATIONS

It is recommended that the stability and control derivatives be calculated for other flight conditions. This could be done in a quick and accurate way using the same programs in Matlab. Some wind tunnel testing would also verify these calculations and explore some non-linear flight conditions like flight at high angles of attack.

The purchase of a portable Unix based workstation would be a wise step towards the integration of the $Matrix_x$/Systembuild as an essential tool for flight testing and design. It is currently necessary to transport a full size Sun workstation to the test site. Doing so because of the large size and the sensitivity of the hardware the equipment is jeopardized during the transport process.

A formal nonlinear parameter estimation approach should be incorporated into the flight test program. However several aspects that currently are not known, such as data acquisition of all desired information, data filtering and sensor behavior should be addressed prior to initiating parameter estimation.

The nonlinear model created in Systembuild is a perfect platform for designing control, guidance and navigation systems which will be implemented, flight tested and improved using the capabilities of $Matrix_x$. Stepping through the rapid prototype design process the design of the controller is significantly simplified.

APPENDIX A: FILES FOR STABILITY AND CONTROL DERIVATIVES

A. FLIGHT CONDITION

```
% File for Frog data which change with flight condition
% frogfc1.m
% Last Update:  02 JAN 97
g = 32.174;                    % Acceleration due to gravity
Wlmg = 30.42;                  % Weight on left main in lbs
Wrmg = 32.13;                  % Weight on right main in lbs
Wng = 5.18;                    % Weight on nose gear in lbs

Umph = 60;                     % Flight speed in miles per hour
Ufps = 88.0;                   % Flight speed in feet per second
rho = .002327;                 % Air density in slugs/(cubic ft)

Ixx = 12.52;                   % Moment of inertia about x-axis
Iyy = 8.43;                    % Moment of inertia about y-axis
Izz = 18.55;                   % Moment of inertia about z-axis
Ixz = 0;                       % Assumed!!!!!!!!!!!!!!!!!!!!!!!

LD =7;                         % Lift to drag ratio

thetanaut =0;                  % Initial pitch angle
```

B. FROG DATA

% File for Frog data which are fixed

% frog.m

% Last Update: 02 Mar 97


```
ac =.271;                    % Aileron chord in ft.#
ai =2.625;                   % distance from centerline to
%                              inner edge of aileron in ft.#
alpha0l = -2*pi/180;         % a.o.a. for zero lift (radians)#
ao = 4.417;                  % distance from centerline to
%                              outer edge of aileron in ft.#
b = 10.58;                   % Span of wing in ft#
bt =3.3125;                  % Span of horizontal tail in ft.#
bv = 1.25;                   % Height of vertical tail in ft.#
cbar = 1.66;                 % Mean aerodynamic chord (m.a.c.)
%                              in feet#
CLalphaafw =5.87649;         % Lift curve slope of wing
%                              airfoil (NACA 2415) in per
%                              radian#
CLalphaaft =5.72958;         % Lift curve slope of horizontal
%                              tail airfoil (NACA 0010-34) in per
%                              radian#
CLalphaafv = 2*pi;           % Lift curve slope of vertical
%                              tail airfoil (flat plate) in per
%                              radian#
CMac = -.045;                % Coefficient of moment about
%                              aero. ctr.#
ct =0.968;                   % m.a.chord of horizontal tail in ft.
```

```
c4tail =5.9086;  %5.8306;      % Location of quarter chord of
%                                horizontal tail in feet from
%                                proptip
c4wing =1.3108; %1.2483;        % Location of quarter chord of
%                                wing in feet from proptip
da0dde = .625;                  % Section flap effectiveness
%                                for 33% flap (elevator)
%                                Abbott and Doenhoff p. 190
da0ddr = .675;                  % Section flap effectiveness
%                                for 38% flap (rudder)
%                                Abbott and Doenhoff p. 190
deda = .4;                      % Downwash angle derivative
%                                estimated from Perkins/Hage
Df = 1.0416;                    % Depth of fuselage in ft.
e0 = 0;                         % Assumed epsilon naught
ee = .8;                        % Assumed span efficiency factor
g = 32.174;                     % gravitational constant
hac = .241;                     % Location in percent chord of
%                                aero. ctr. (NACA 2415)
it = (4.5+2)*pi/180;            % Incidence angle of hor. tail
lewing =.8958;                  % Location of leading edge of wing
%                                in feet from proptip
letail =5.667;                  % Location of leading edge of
%                                horizontal tail in feet from
%                                proptip
mg =19.5/12;                    % Location of main gear in ft
%                                from proptip
ng =-5/12;                      % Location of nose gear in ft
```

```
%                              from proptip
S = 17.5;                      % Reference (wing) area in sq. ft.
Sr = .501736;                  % Rudder area in sq. ft.
St =3.17448;                   % Horizontal tail area in sq. ft.#
Sv =0.98177;                   % Vertical tail area in sq. ft.#
Wf = .75;                      % Width of fuselage in ft.#
ybar = b/4;                    % Spanwise location of m.a.c.#
zv = .416;                     % Vert. tail height to m.a.c.
%                               (estimated)
Zwf = .5833;                   % Vertical height of wing
%                                above fuselage C.L. in ft.
```

C. PHYSICAL CALCULATIONS

```
% File to calculate physical considerations
% physfrog.m
% Last Update:  04 FEB 97


% Load frog data
frog
% Load flight condition data
fcfrog


% Calculate aircraft weight
W = Wlmg + Wrmg + Wng;


% Calculate aircraft mass
m = W/g;


% Calculate aspect ratio of wing
AR = b^2/S;


% Calculate aspect ratio of hor. tail
ARt = bt^2/St;


% Calculate aspect ratio of vert. tail
ARv = bv^2/Sv;


% Calculate longitudinal center of gravity
h = ((ng*Wng + mg*(Wlmg+Wrmg))/W-lewing)/cbar;
```

```
% Calculate "tail length" from c.g. to horizontal tail a.c.
% same for horizontal and vertical
lt = c4tail - (lewing + h*cbar);
lv = lt;


% Calculate "tail length" from c/4 wing to c/4 tail
ltprime = c4tail - c4wing;


% Calculate hor. tail volume coefficient
VH = lt*St/(S*cbar);


% Calculate vert. tail volume coefficient (yaw)
VV = lv*Sv/(b*S);


% Calculate vert. tail volume coefficient (roll)
Vprime = zv*Sv/(b*S);


% Unit antisymmetrical angle of attack for outer and inner
% edge of aileron (See Smetana p. 141)
antisymo = ao/(b/2);% 0.83497
Cldatauo = .625;
antisymi = ai/(b/2);% 0.49622
Cldataui = .248;
cacw = ac/cbar;%0.16325
tau = .48;


% for yawing moment due to aileron, see p. 142, Smetana
eta = ai/(b/2);%0.49622  AR=6.3963
```

K = -.175;

% for side force due to rudder deflection, see Smetana p. 145
vratio = Sr/Sv;% 0.511052
taur = .675;

% for rolling moment due to sideslip, See Raymer, Fig. 16.21, p. 439
ClBwCL = -.04;

D. NONDIMENSIONAL DERIVATIVES CALCULATIONS

```
% File to calculate nondimensional derivatives
% ndderiv.m
% Last Update:  12 FEB 97

% Load Frog data with flight condition
phfrog

% Calculate coefficients of lift and drag
CL = W/(.5*rho*Ufps^2*S);
CD = CL/LD;
D=CD*(.5*rho*Ufps^2*S);

% Calculate lift curve slope of wing in per radian
CLalphaw = CLalphaafw/(1+CLalphaafw/(pi*ee*AR));

% Calculate lift curve slope of horizontal tail in per radian
CLalphat = CLalphaaft/(1+CLalphaaft/(pi*ee*ARt));

% Calculate lift curve slope of vertical tail in per radian
CLalphav = CLalphaafv/(1+CLalphaafv/(pi*ee*ARv));

% Calculate change in hor. tail lift with change in elevator
dcLtdde = da0dde * CLalphat;        % per radian

% Calculate change in vert. tail lift with change in rudder
dcLvddr = da0ddr * CLalphav;        % per radian
```

```matlab
% Calculate zero lift pitching moment
CM0 = CMac + VH * CLalphat * (it + e0);


% Calculate CMalpha in per radian
CMalpha = CLalphaw*((h-hac)-VH*(CLalphat/CLalphaw)*(1-deda));


% Calculate change in aircraft lift with change in elevator
CLdelta = dcLtdde*(St/S);     % per radian
   ,

% Calculate change in aircraft pitching moment w. chng in elevator
CMde = -1*VH*dcLtdde;     % per radian


% Calculate angle of attack and elevator angle for trimmed flight
%
%      CM = CM0 + CMalpha*alpha + CMde*de
%      Cl = CLalphaw*alpha + CLdelta*de
%
%
%      _          __  _   _    _   _
%     | CLalphaw   CLdelta | | alpha |    | CL   |
%     |                    | |       | = |      |
%     |_CMalpha    CMde___| |__ de _|    |_-CM0_|
%
%           A      *   X   =    C
%
A = [ CLalphaw  CLdelta
    CMalpha    CMde   ];
C = [    CL
    -1*CM0 ];
```

```
X = inv(A)*C;
atrim = X(1,1)          % trim a.o.a. in radians
etrim = X(2,1)          % trim elevator in radians


% Calculate change in yawing moment with change in rudder
% "rudder power"
% assumes VF/Vinfinity = 1
Cndr = -1*VV*dcLvddr;      % in per radian


% Calculate CnB contribution from vert. tail
% CnB = CLalphav*VV*(VF/Vinfinity)^2*(1-dsigma/dbeta)
% assumes VF/Vinfinity = 1 and dsigma/dbeta = 0
CnB = CLalphav*VV;% in per radian


% Calculate change in rolling moment with change in sideslip


% First calculate dihedral contribution from wing
% Raymer p. 439
ClBwf = -1.2*sqrt(AR)*Zwf*(Df+Wf)/b^2;
ClBw = ClBwCL*CL+ClBwf;


% Next calculate contribution from fin
% ClBv = -1*Clalphav*Vprime*(VF/Vinfinity)^2*(1-dsigma/dbeta)
% Assume VF/Vinfinity = 1 and dsigma/dbeta = 0
ClBv = -1*CLalphav*Vprime;       % in per rad


% Combine ClBg and ClBv into ClB
ClB = ClBw + ClBv;   % in per rad
```

```
% Calculate "aileron power", Clda
% See Smetana pp. 139-141
Cldatau = Cldatauo - Cldataui;
Clda = Cldatau*tau;    % in per radian


% Calculate change in yawing moment w. aileron deflection
Cnda = 2*K*CL*Clda;          % in per radian


% Calculate side force due to yaw
% By Smetana p. 107
CyB = -.31;     % in per radian


% Calculate side force due to rudder
Cydr = CLalphav*taur*Sv/S; % in per radian


% Calculate side force due to aileron
% By Smetana, p. 138
Cyda = 0;


% Calculate rolling moment due to rudder
Cldr = Cydr*zv/b;      % in per radian


% Calculate change in drag due to change in elevator
% Smetana pp. 95-100
% Using Figure 25 at 6 degrees aoa
CDde = ((.16-.03)/(20*pi/180))*St/S;          % in per radian
```

% Calculate change in drag with change in aoa

% Smetana pp. 64-65

% Assuming dCD0/dalpha is negligible

CDalpha = 2*CL*CLalphaw/(pi*ee*AR);   % in per radian

% Calculate change in pitching moment w.r.t. alphadot

% Smetana pp. 78-81, etat assumed = 1

CMalphadot = -2*CLalphat*deda*(ltprime/cbar)*...

(lt/cbar)*(St/S);   % in per radian

% Calculate change in lift with pitch rate

% Smetana pp. 82-85

% Neglecting wing contribution, assuming etat = 1

CLq = 2*(lt/cbar)*CLalphat*(St/S); % in per radian

                              .

% Calculate change in lift with alphadot

% Smetana pp. 75-76

CLalphadot = -1*CMalphadot/(lt/cbar);   % in per radian

% Calculate change in pitching moment w. pitch rate

% Smetana pp. 87-88

% Assuming etat = 1

CMq = -2*(cbar/4-h*cbar)*abs(cbar/4-h*cbar)*CLalphaw/(cbar^2) - ...

2*(lt/cbar)^2*CLalphat*(St/S);   % in per radian

% Calculate roll damping

% Smetana pp. 122-125 % Clp(a0:2pi)=-0.475

% Neglecting contribution from vertical tail

Clp = -.475*(AR+4)/(2*pi*AR/CLalphaw+4);   % in per radian

% Calculate change in yawing moment due to rolling

% Smetana pp. 126-129

% Neglecting contribution from vertical tail

Cnp = -1*CL/8;   % in per radian

% Calculate change in side force with yaw rate

% From Schmidt p. 3-23

% Assume etat = 1

Cyr = 2*VV*CLalphav;   % in per radian

% Calculate change in rolling moment w. yaw rate

% Schmidt p. 3-24

% Tail contribution

Clrv = (zv/b)*Cyr;   % in per radian

% Wing contribution

Clrw = CL/4;   % in per radian

% Total

Clr = Clrv + Clrw;   % in per radian

% Calculate yaw damping

% Schmidt p. 3-25

% Tail contribution

Cnrv = -1*(lv/b)*Cyr; % in per radian

% Wing contribution from Smetana p. 136

CD0 = CD-CL^2/(pi*ee*AR);

Cnrw = -.02*CL^2-.3*CD0;   % in per radian

% Total

Cnr = Cnrv + Cnrw;   % in per radian

% The following 3 derivatives are negligible and taken to be 0

CDq = 0;   % in per radian

Cyq = 0;   % in per radian

Cyp = 0;   % in per radian

% A few misc. calculations

% Static Margin/Neutral Point

statmar = CMalpha/(-1*CLalphaw);

hn = statmar + h

E. DIMENSIONAL DERIVATIVES CALCULATIONS

```
% File to calculate dimensional derivatives

% frogdder.m

% Last Update:  12 FEB 97


% Run nondimensional derivative program
ndfrog


% Calculate dynamic pressure
qbar = .5*rho*Ufps^2;                   % ft lbs


Malpha = CMalpha*qbar*S*cbar/Iyy;       % per second^2


Mq = CMq*(cbar/(2*Ufps))*qbar*S*cbar/Iyy;
                                        % per second


Malphadot = CMalphadot*(cbar/(2*Ufps))*qbar*S*cbar/Iyy;
                                        % per second


Xu = -2*CD*qbar*S/(m*Ufps);             % per second


Zu = -2*CL*qbar*S/(m*Ufps);             % per second


Zalphadot = CLalphadot*(cbar/(2*Ufps))*(qbar*S/m);
                                        % ft per second


Zq = CLq*(cbar/(2*Ufps))*(qbar*S/m);    % ft per second
```

```
Mu = 0;                                    % per ft second

Xde = -1*CDde*qbar*S/m;                    % ft per second^2

Zde = -1*CLdelta*qbar*S/m;                 % ft per second^2

Mde = CMde*qbar*S*cbar/Iyy;                % per second^2

Xalpha = (CL - CDalpha)*qbar*S/m;          % ft per second^2

Zalpha = -1*(CLalphaw+CD)*qbar*S/m;        % ft per second^2

YB = CyB*qbar*S/m;                         % ft/second^2

LB = ClB*qbar*S*b/Ixx;                     % 1/second^2

NB = CnB*qbar*S*b/Izz;                     % 1/second^2

Yp = Cyp*b*qbar*S/(2*Ufps*m);             % ft/sec

Yr = Cyr*b*qbar*S/(2*Ufps*m);             % ft/sec

Lp = Clp*(b/(2*Ufps))*qbar*S*b/Ixx;       % 1/sec

Np = Cnp*(b/(2*Ufps))*qbar*S*b/Izz;       % 1/sec

Lr = Clr*(b/(2*Ufps))*qbar*S*b/Ixx;% 1/sec
```

```
Nr = Cnr*(b/(2*Ufps))*qbar*S*b/Izz;        % 1/sec


Ydr = -1*Cydr*qbar*S/m;                      % ft/sec^2


Yda = 0;                                     % ft/sec^2


Ldr = Cldr*qbar*S*b/Ixx;                     % 1/sec^2


Lda = Clda*qbar*S*b/Ixx;                     % 1/sec^2


Ndr = Cndr*qbar*S*b/Izz;                     % 1/sec^2


Nda = Cnda*qbar*S*b/Izz;                     % 1/sec^2


Malphaprime = Malpha + Malphadot*(Zalpha/Ufps);
Mqprime = Mq + Malphadot;
Mdeprime = Mde + Malphadot*(Zde/Ufps);
```

F. STABILITY AND CONTROL DERIVATIVES

% File to get values of dimensional and nondimensional derivatives

% Last Update: 04 FEB 97

% Run programs to calculate derivatives

ddfrog

% Nondimensional Derivatives

CL

CD

CD0

CLalphaw

CMalpha

CDalpha

CLalphadot

CMalphadot

CLq

CMq

CLdelta

CDde

CMde

CyB

CnB

ClB

Cyp

Cnp

Clp

Cyr

Cnr

Clr

Cyda

Cnda

Clda

Cydr

Cndr

Cldr

CDq

Cyq

% Longitudinal Dimensional Derivatives

Xu

Xalpha

Xde

Zu

Zalpha

Zalphadot

Zq

Zde

Mu

Malpha

Malphadot

Mq

Mde

% Lateral/Directional Dimensional Derivatives

YB

Yp

Yr

Yda

Ydr

LB

Lp

Lr

Lda

Ldr

NB

Np

Nr

Nda

Ndr

APPENDIX B: MOMENTS OF INERTIA CALCULATIONS

For the formula:

$$I = \frac{[W_M + 2\omega\,(L_{SHORT} + L_{LONG})]\bar{Z}_{M+S}}{4\pi^2} P_{M+S}^2 - \frac{W_M \bar{Z}_M^2}{g} - \frac{2\omega}{3g}(L_{SHORT}^3 + L_{LONG}^3)$$

the following data were used for the calculation of the moments of inertia.

Constant values:

Weight of the model,	$W_M = 67.73$ lbs.
Weight of the unit length chain,	$\omega = 0.061887$ lbs/ft.
Length of short chain,	$L_{SHORT} = 13.5$ ft.
Length of long chain,	$L_{LONG} = 15$ ft.
Gravitational constant,	$g = 32.1472$ ft/s^2.

(adjusted for latitude and elevation)

Variable values:

I_{XX}: Distance from pivot to center of gravity
of model and support, $\bar{Z}_{M+S} = 13.91166$ ft

Distance from pivot to center of gravity $\bar{Z}_M = 14$ ft.
of model,

Average period of model and support, $P_{M+S} = 4.18475$ sec.

I_{YY}: Distance from pivot to center of gravity
of model and support, $\bar{Z}_{M+S} = 13.91166$ ft.

Distance from pivot to center of gravity
of model, $\bar{Z}_M = 14$ ft.

Average period of model and support, $P_{M+S} = 4.16475$ sec.

I_{ZZ}: Distance from pivot to center of gravity
of model and support, $\bar{Z}_{M+S} = 13.41775$ ft.

Distance from pivot to center of gravity
of model, $\bar{Z}_M = 13.5$ ft.

Average period of model and support, $P_{M+S} = 4.1455$ sec.

APPENDIX C: MATLAB FILES OF EQUATIONS OF MOTION

A. ABMAT.M

% File for modelling the nonlinear equations of motion

% Last Update: 21 JAN 1997

function accel = abmat(x)

% calculates the accelerations (angular and linear) due to

% aero forces; w X v; gravity.

% Variables brought from workspace:

% x = combination vector [contrl inputs, state variables, euler angles]

% (da, de, dr, dtrt, u, v, w, p, q, r, phi, theta, psi)

% Variables called from function "frog_data"

% rho = air density

% b = wing span

% c = wing cord

% s = wing area

% Cfo = Steady state force term

% Cfu = Stability derivative for control inputs

% m = airplane mass

% Ib = inertia tensor matrix (body frame)

% To = Thrust scale term

% Pe = Engine position matrix

% get the aircraft data

```
[u0,w0,rho,Cfx,Cfo,Cfu,Cfxdot,s,b,c,m,Pe,To,Ib] = frog_data;
% seperate the combined vector into seperate elements


u = [x(1); x(2); x(3)];
dtrt = x(4);
state = [x(5); x(6); x(7); x(8); x(9); x(10)];
lambda = [x(11); x(12); x(13)];


%%%%%%  calculate velocity wrt the airmass and form state vector
%%%%%%  that will be used to calculate the aerodynamic forces/moments


state1 = [state(1)-u0; state(2); state(3)-w0; x(8); x(9); x(10)];


%%%%%%  calculate total velocity, vt


vt = (state(1)^2 + state(2)^2 + state(3)^2)^.5;


% calculate qbar


qbar = .5*rho*(vt^2);


%  calculate M1


M1 = diag([1/vt, 1/vt, 1/vt, (.5*b)/vt, (.5*c)/vt, (.5*b)/vt]);


%  calculate M2


M2 = diag([0, 0, (.5*c)/(vt^2), 0, 0, 0]);
```

72

```
% calculate Sprime

Sprime = diag([-1, 1, -1, b, c, b]*s);

% calculate Mu

Mu = inv([eye(3)*m,zeros(3);zeros(3),Ib]);

% calculate Tw2b

alpha = state(3)/vt;
beta = state(2)/vt;
T1 = [cos(alpha), 0, sin(alpha); 0,1,0; -sin(alpha), 0, cos(alpha)];
T2 = [cos(beta), -sin(beta), 0; sin(beta), cos(beta), 0; 0,0,1];
Tw2b = [T1'*T2', zeros(3); zeros(3), T1'*T2'];

% calculate Chi

Chi = eye(6) - Mu*Tw2b*qbar*Sprime*Cfxdot*M2;

% calculate Propulsion matrix

Prop = [ eye(3);
0,-Pe(3),Pe(2);
Pe(3),0,-Pe(1);
-Pe(2),Pe(1),0];
```

% calculate gravity vector and rotation matrix

```
Rot = [1, 0, -sin(lambda(2));
0, cos(lambda(1)), cos(lambda(2))*sin(lambda(1));
0, -sin(lambda(1)), cos(lambda(2))*cos(lambda(1))];

Ru2b = [Rot;zeros(3)];

g = [0; 0; 32.174];
FgU = m*g;
```

% put the components due to gravity; thrust; and control surface deflections
% together for their contribution to x-dot

```
thrust = Prop*To*dtrt;
gravity = Ru2b*FgU;
ctrl=qbar*(Tw2b*(Sprime*(Cfo + (Cfu*u))));

xdotu=(Mu*(ctrl+thrust+gravity));
```

% calculate rotation matrix

```
omegax = [0,-state(6),state(5);state(6),0,-state(4);-state(5),state(4),0];
wxIb = (-inv(Ib))*(omegax*Ib);

Rot = [-omegax, zeros(3); zeros(3), wxIb];
```

% rotation component of xdot (w X v)

```matlab
xdotrot = Rot*state;

%  state vector feedback component xdot

xdotcfx =qbar*(Mu*(Tw2b*(Sprime*(Cfx*(M1*state1)))));

% add three components of xdot together and premult by inv(Chi)

xdot= inv(Chi)*(xdotrot+xdotcfx+xdotu);

% calc accel that a strap-down IMU would measure

xdotb=xdot-xdotrot-Ru2b*g;

% put together for the return vector

%accel=[xdotb(1);xdotb(2);xdotb(3);xdot];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

% return xdot only

accel=xdot;
```

B. FROG_DAT.M

% File for inputting all aircraft data for equations of motion

% Last Update: 21 JAN 1997

function [u0,w0,rho,Cfx,Cfo,Cfu,Cfxdot,s,b,c,m,Pe,To,Ib] =frog_data

u0 = 88;

w0 = 0;

% Sea level- std day

rho = .0023769;

% derivative matrix due to state variables

% rows: [CD CY CL Cl Cm Cn]

% col: [u v w p q r]

```
Cfx = [0   0 .23 0 0 0;
       0 -.31 0 0 0 .1151;
       0   0 4.3034 0 3.35 0;
       0 -.0509 0 -.3702 0 .1119;
       0 0 -0.5565 0 -8.8818 0;
       0 .0575 0 -.0537 0 -.0669];
```

% derivative matrix due to control inputs

% rows: [same as above]

% col: [elev rud ail]

```
Cfu = [.0676 0 0;
0 .0926 0;
.3914 0 0;
0 .0036 .1810;
-1.0469 0 0;
0 -.0388 -.0272];

% derivative matrix due to x-dot

Cfxdot = [0 0 0 0 0 0;
0 0 0 0 0 0;
0 0 1.3877 0 0 0;
0 0 0 0 0 0;
0 0 -3.7115 0 0 0;
0 0 0 0 0 0];

% steady state force vector

Cfo = [.0614; 0; .4295; 0; 0; 0];

% physical dim.
% WT =67.73 LBS.

m = 2.1051;
s = 17.5;
b = 10.58;
c = 1.66;
```

% engine data

Pe = [17.63/12; 0; -14.92/12];

To = [9.6757 ;0;0];

% inertia tensor matrix

Ib = [12.52 0 0; 0 8.43 0; 0 0 18.55];

APPENDIX D: NUMERICAL LINEARIZATION RESULTS

% Numerical results from linearization of equations of motion

% Last Update: 20 FEB 97

% The state vector at trim will be:

x =[87.9999; 0.0000; 0.1593; 0.0000; 0.0000; 0.0000; 0.0000; 0.0018; 0.0000]'

% the inputs at trim are:

u =[-0.0431; 0.0000; 0.0000; 0.9805]'

% while the outputs are as defined:

y =[0.0000; 88.0000]'

% Then the matrices of the linear model are the following:

A=[-0.1014   0.0000   0.1722   0 -0.1532   0   0.0000   -32.1739 0
      0.0000 -0.2183   0.0000   0.1593 0   -87.4705 32.1739   0.0000   0
     -0.7162   0.0000 -3.7510   0   84.6195   0 -0.0002 -0.0578   0
      0.0000 -0.0682   0.0000   -3.0280 0.0000   0.9165 0.0000   0.0000   0
      0.0412   0.0000 -0.1532   0   -3.7244   0   0.0000   0.0007   0
      0.0000   0.0599   0.0000   -0.3002 0.0000 -0.3683 0.0000   0.0000   0
      0   0   0   1.0000 0.0000   0.0018 0.0000   0.0000   0
      0   0   0   0   1.0000   0.0000 0.0000   0   0
      0   0   0   0   0.0000   1.0000 0.0000   0.0000   0]

B =[-5.1184   0   0   4.5963

      0.0000   7.0847   0   0

    -29.6178   0   0   0

      0.0000   0.4995 24.6412 0

    -32.8288   0   0   -1.4271

      0.0000   -3.5636 -2.4685   0

          0   0   0   0

          0   0   0   0

          0   0   0   0]

C=[0.0000   0 -0.0114   0   0   0   0   1.000   0

    1.0000   0.0000   0.0018   0   0   0   0   0   0]

D=[0   0   0   0 ; 0   0   0   0]

% The eigenvalues of the A matrix are:

Eigenvalue	Damping	Freq. (rad/sec)
0.0474	-1.0000	0.0474
0	-1.0000	0
-0.0293 + 0.5597i	0.0522	0.5604
-0.0293 - 0.5597i	0.0522	0.5604
-0.1964 + 2.4972i	0.0784	2.5049
-0.1964 - 2.4972i	0.0784	2.5049
-3.2691	1.0000	3.2691
-3.7591 + 3.5964i	0.7226	5.2024
-3.7591 - 3.5964i	0.7226	5.2024

E. ADDITIONAL SUPERBLOCK DIAGRAMS
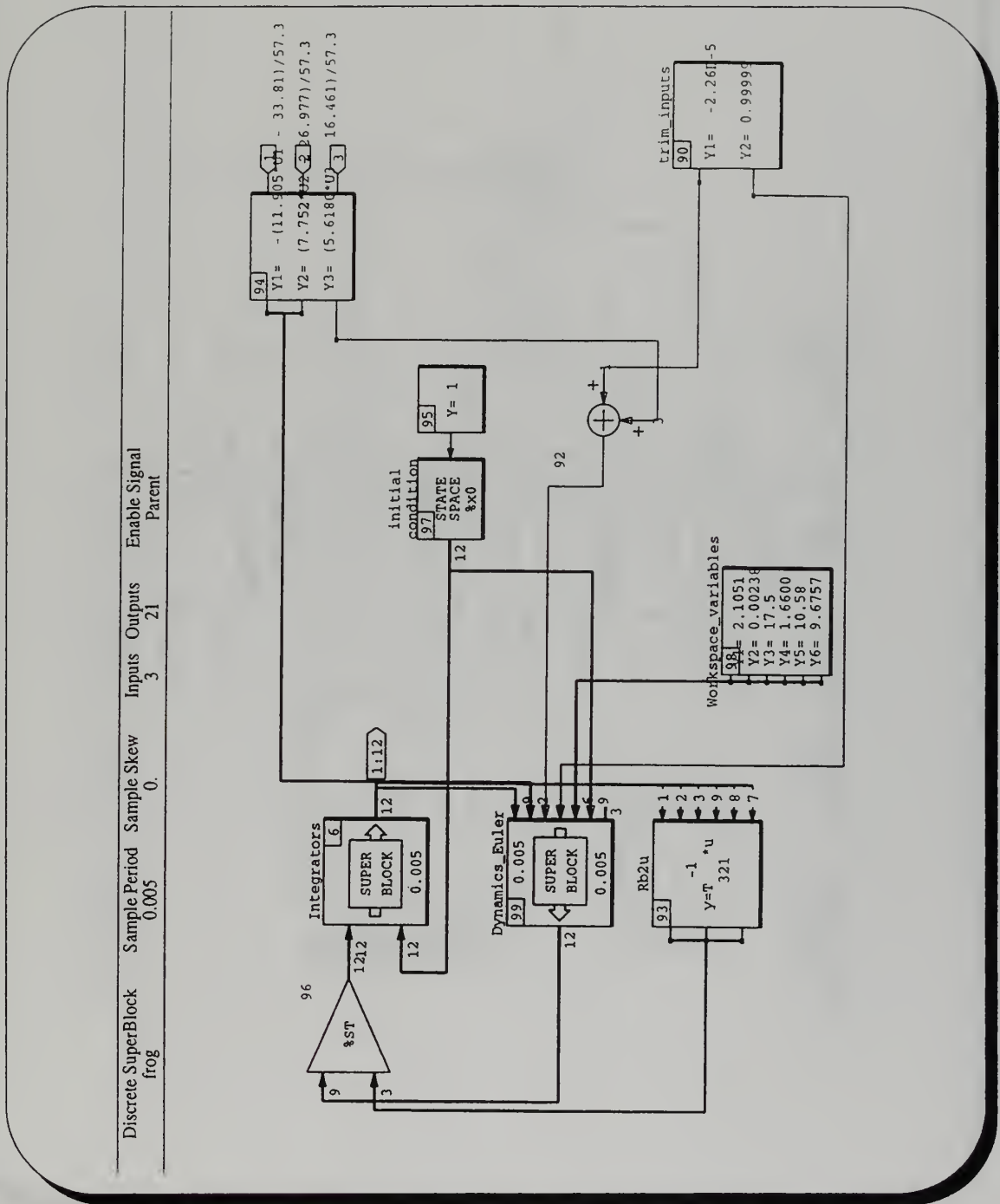
Fig. E.1 Frog superblock diagram
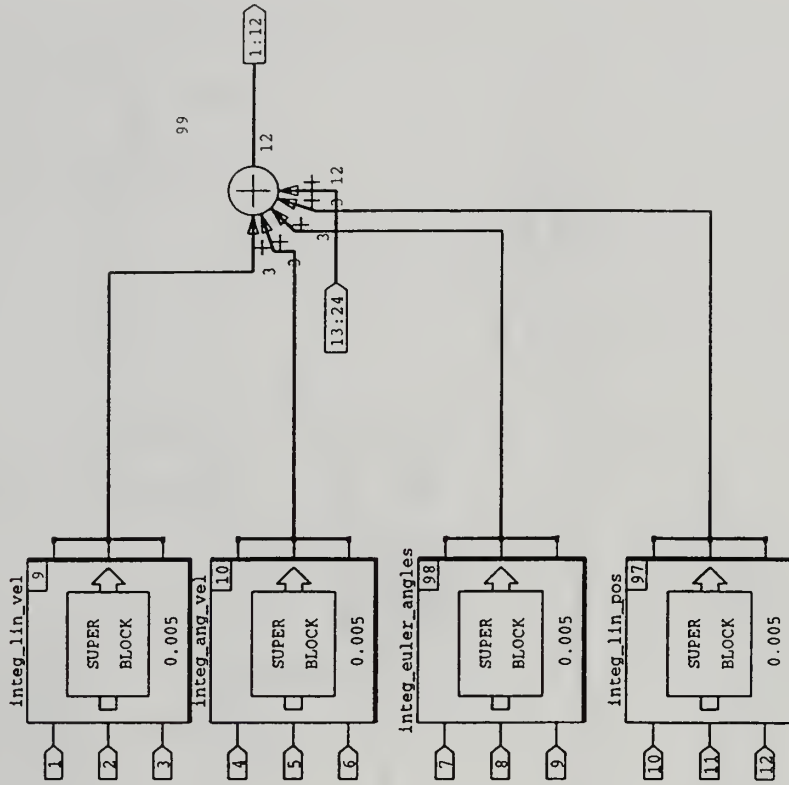
Fig. E.2 Integrators Superblock

82
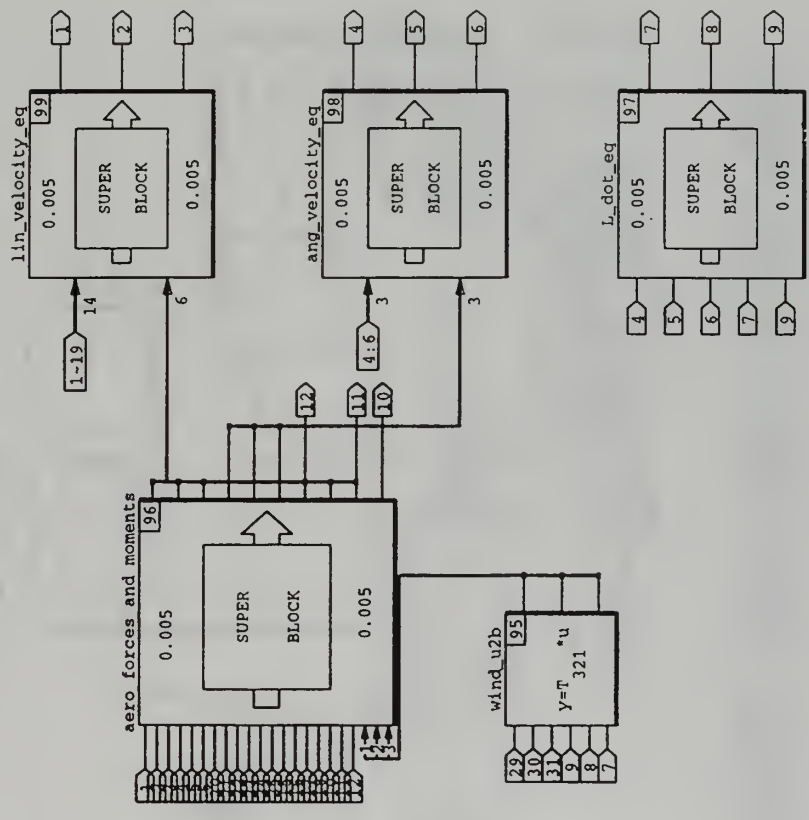
Fig. E.3 Dynamics_Euler Superblock

83

Fig.E.4 Aero_forces_and_moments Superblock

84
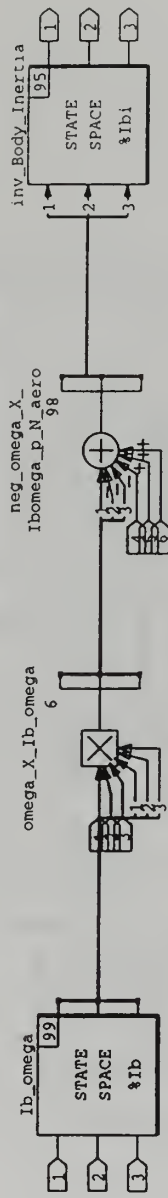
Fig. E.5 Ang_velocity_eq Superblock

85

Fig. E.6 L_dot_eq

86
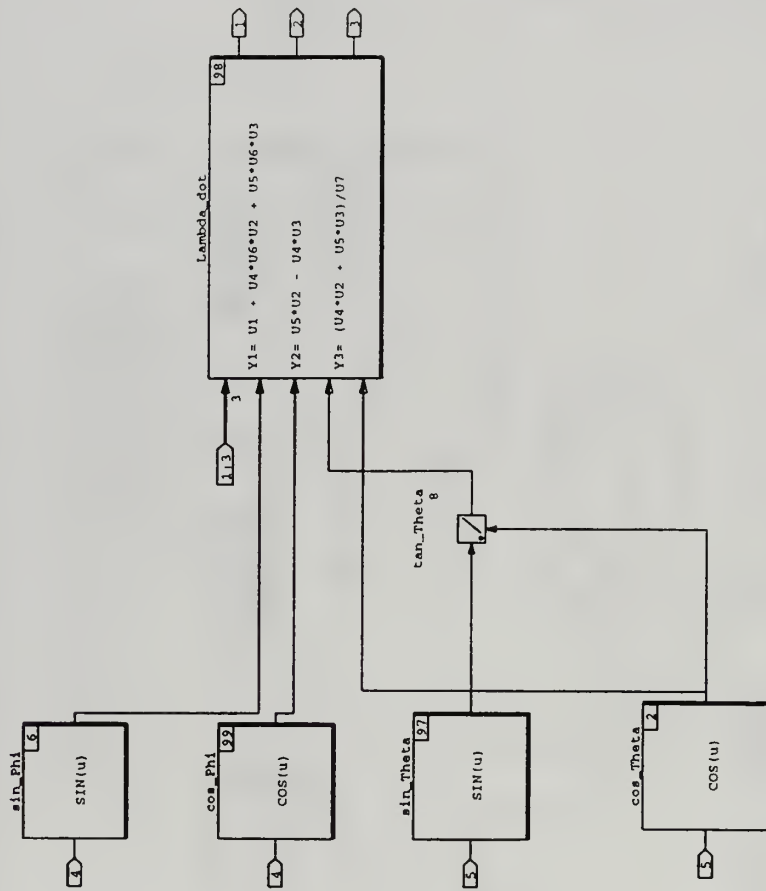
APPENDIX F: FLIGHT TEST RESULTS
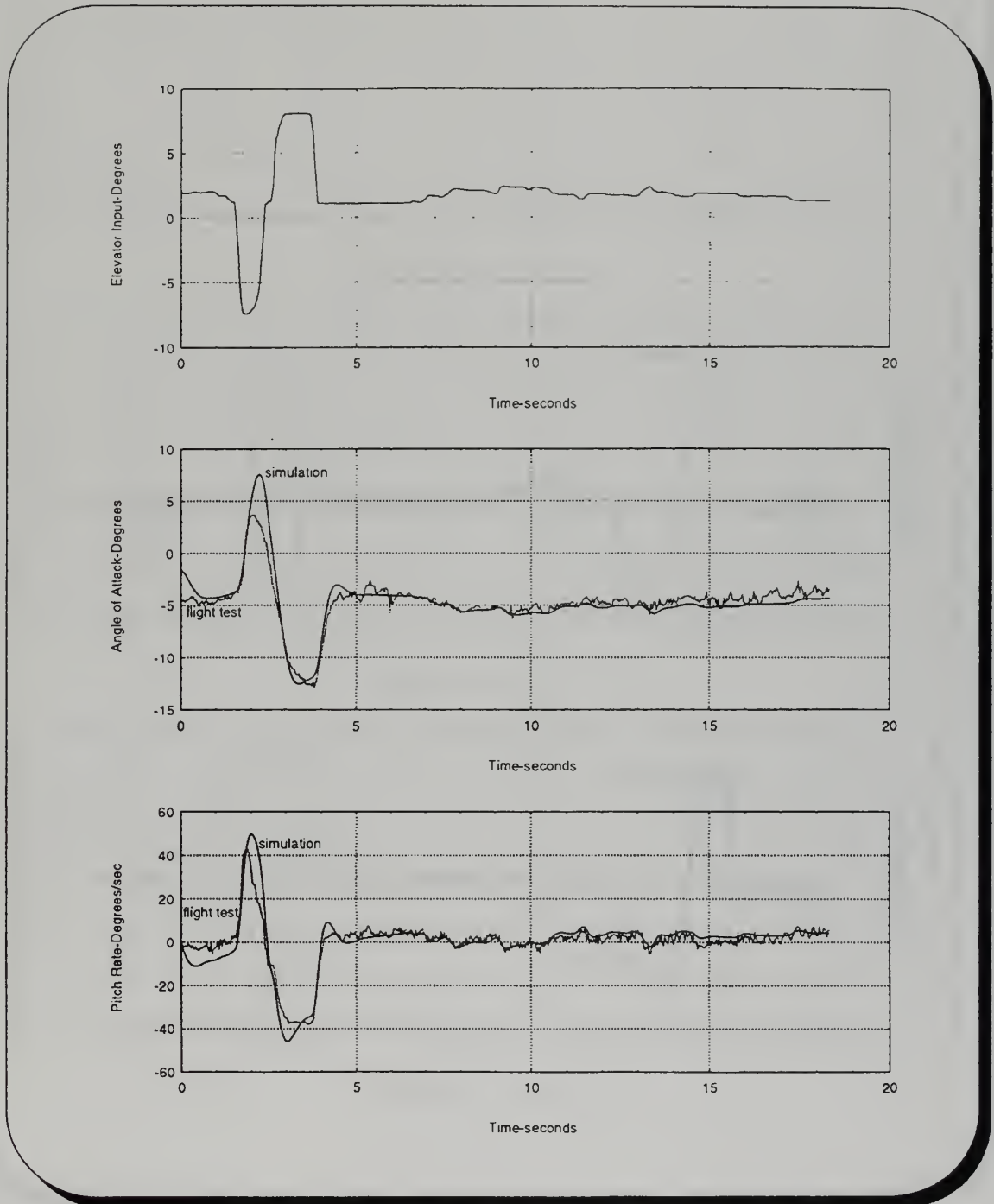
Fig. F.1 Short Period Response- Run #1

87
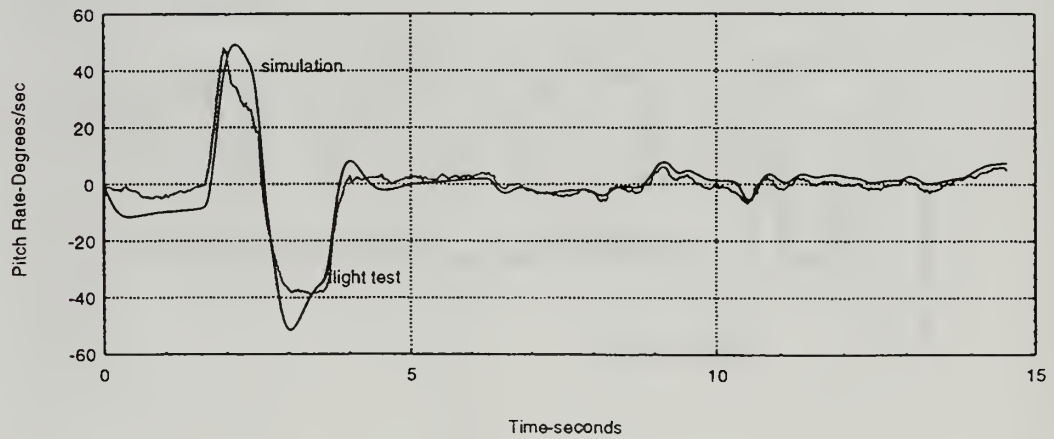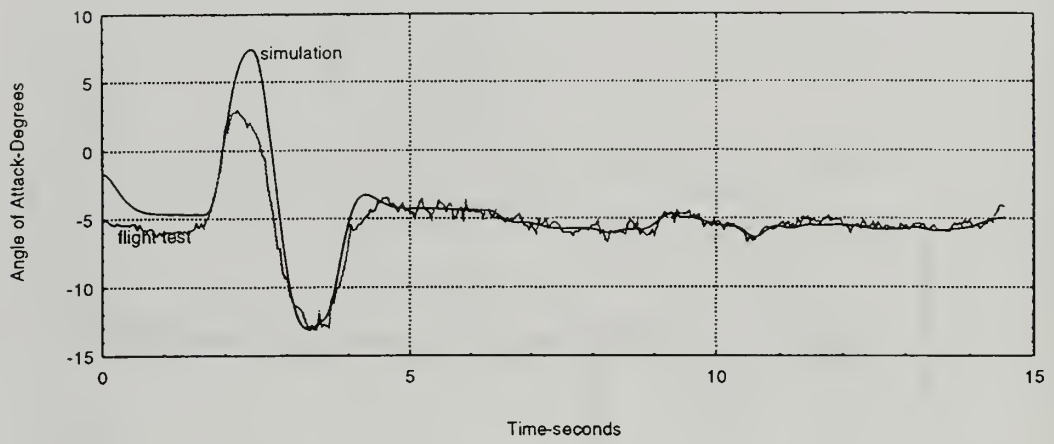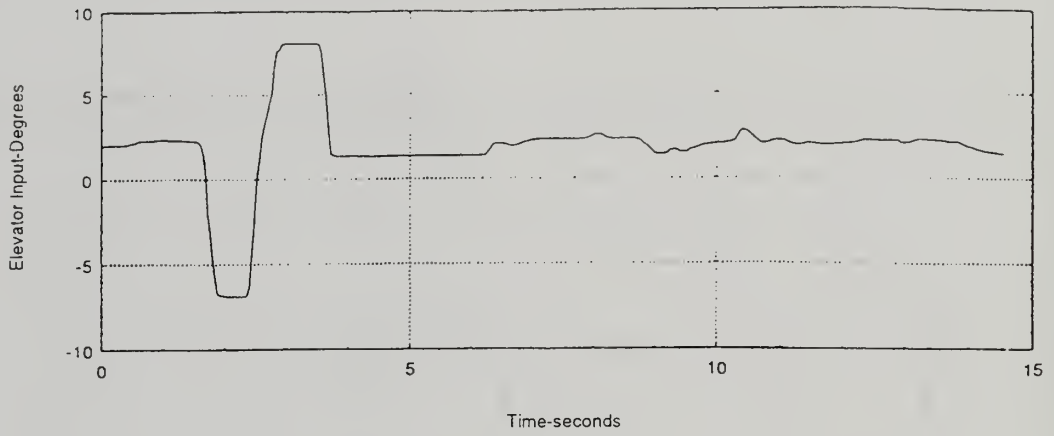
Fig. F.2 Short Period Response- Run #2

88
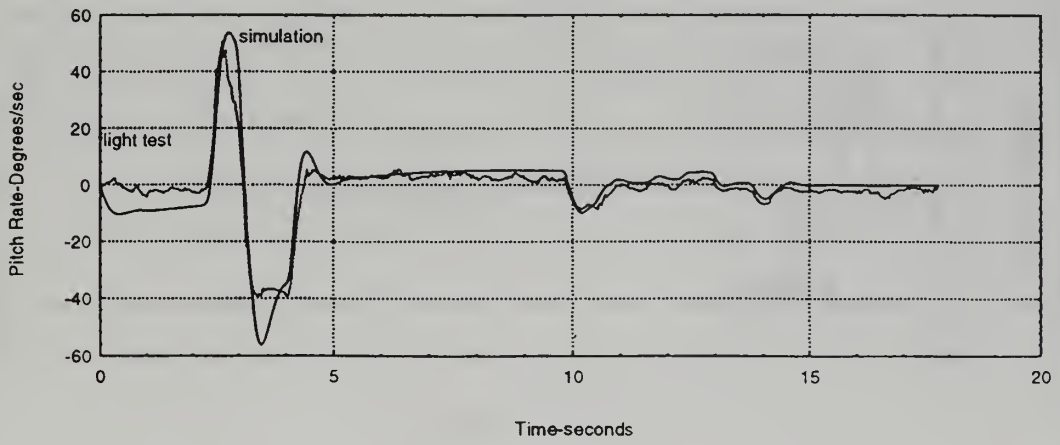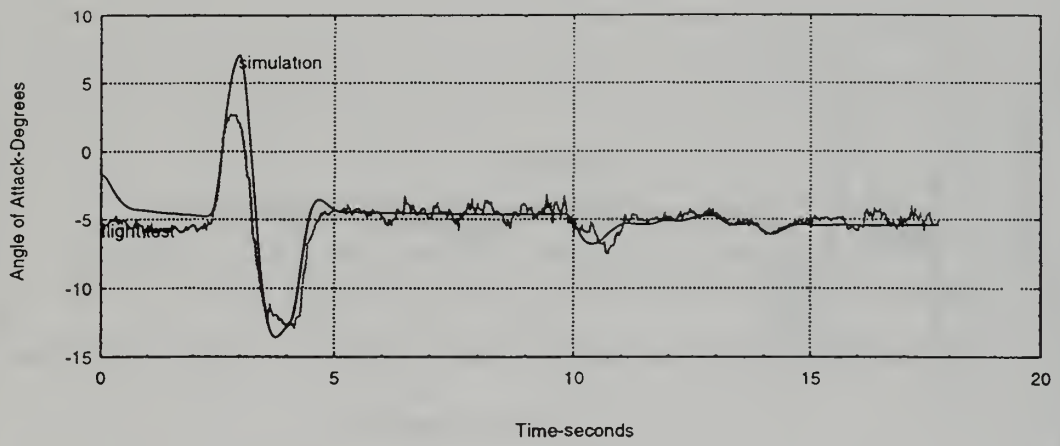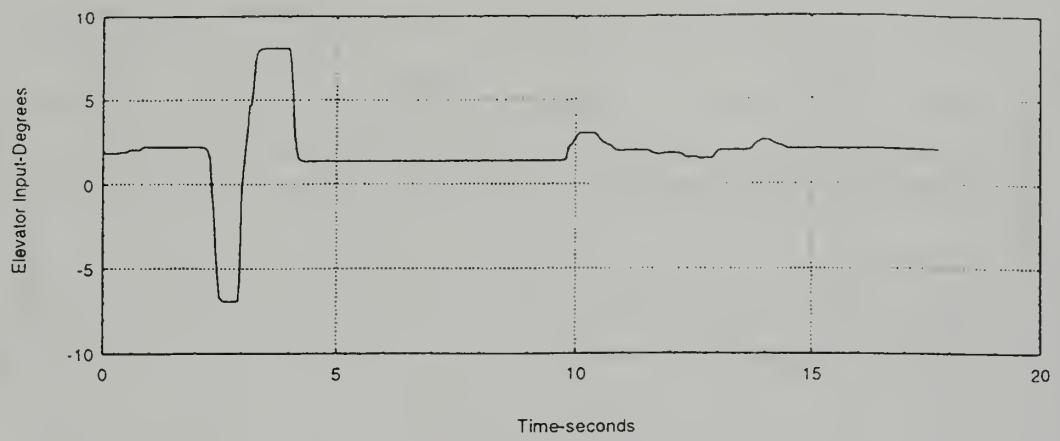
Fig. F.3 Short Period Response- Run #3

89
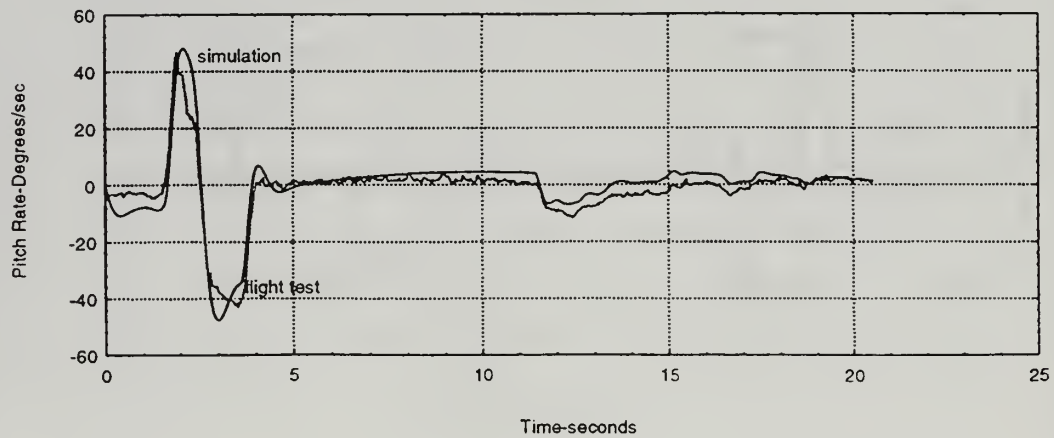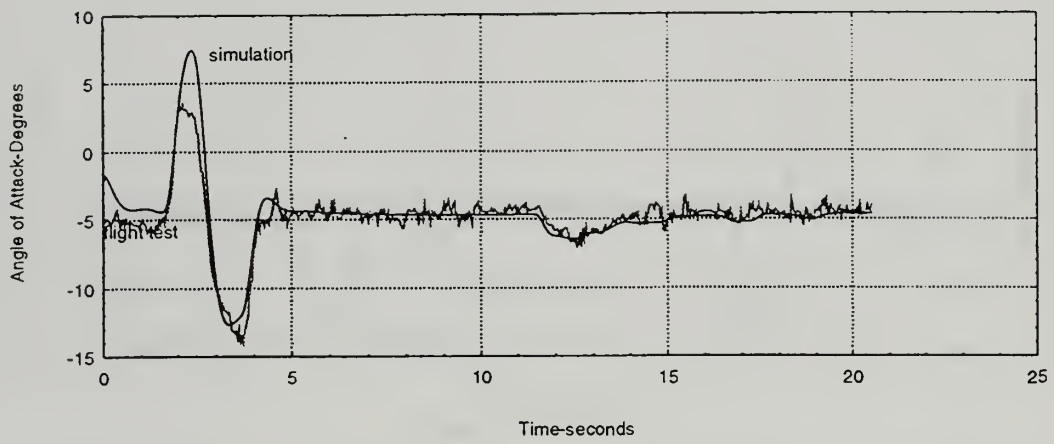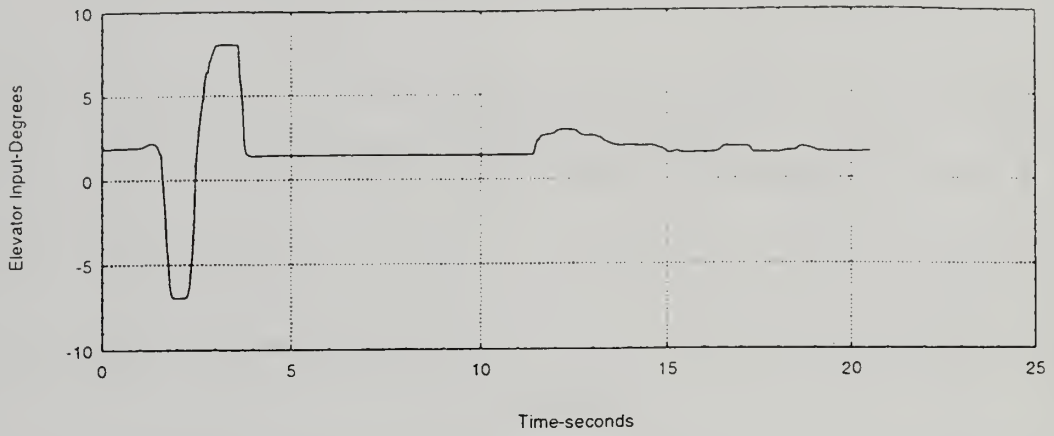
Fig. F.4 Short Period Response- Run #4

90
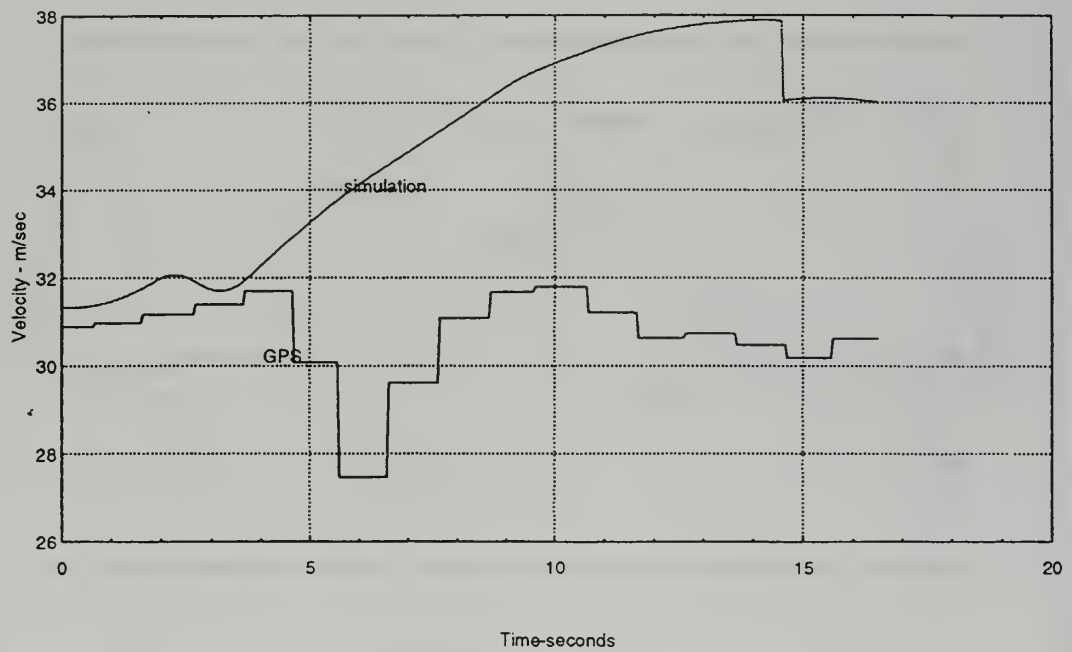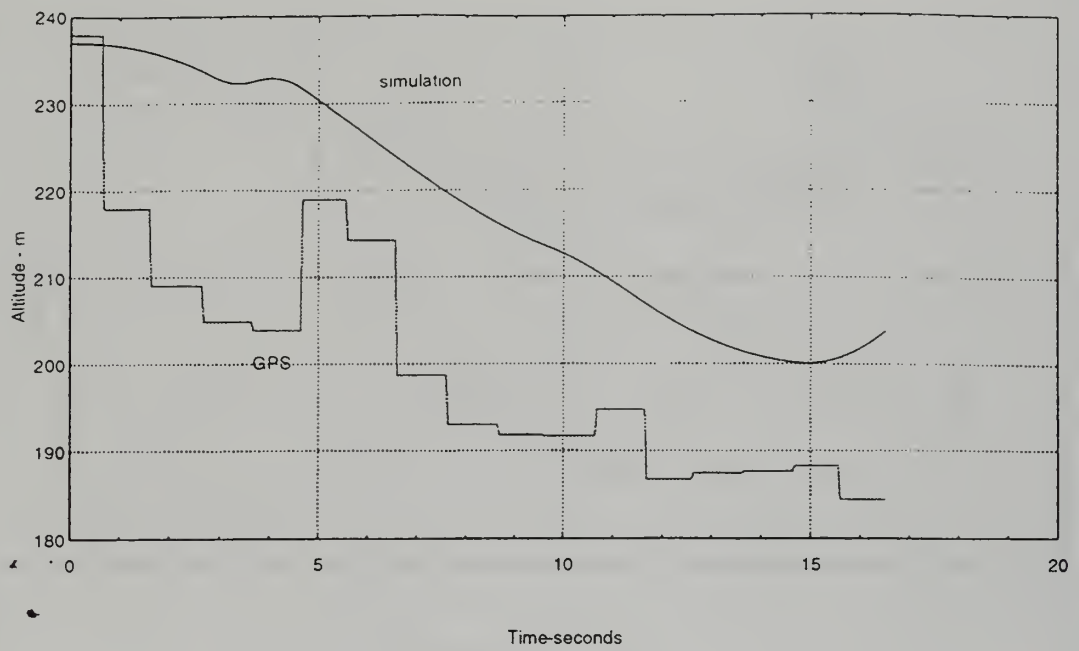
Fig. F.5 Phugoid Response- Run #1

91
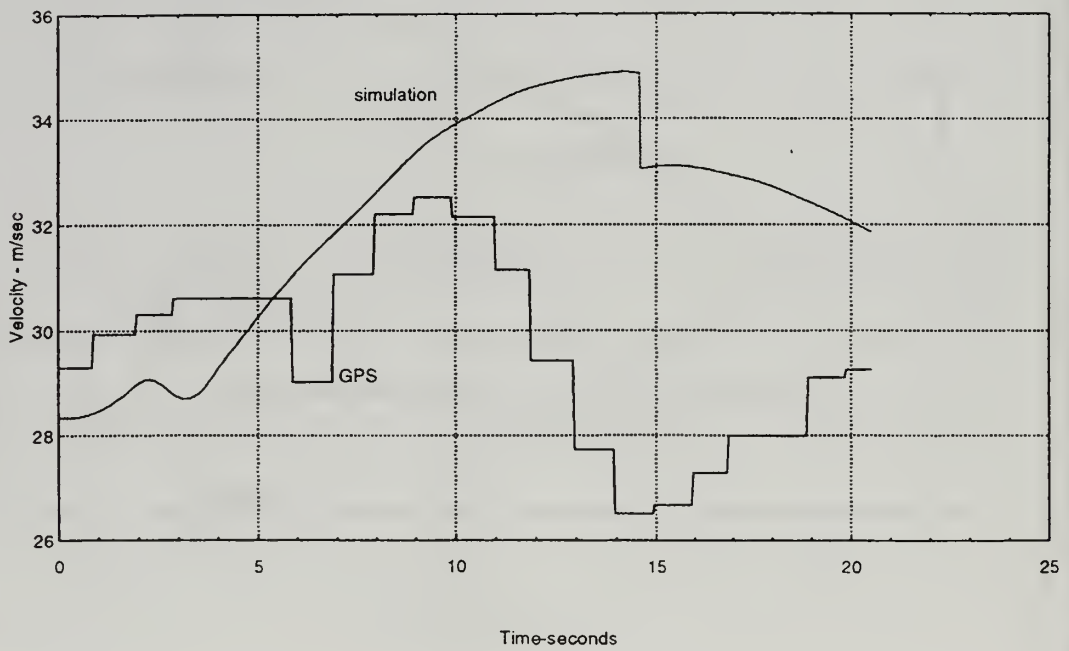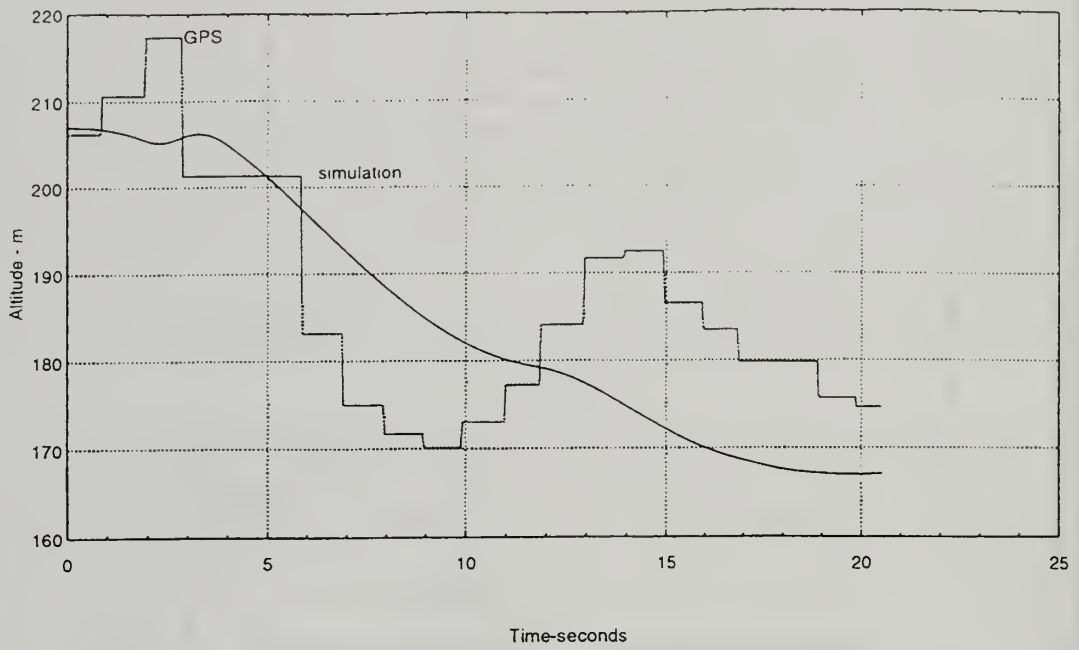
Fig. F.6 Phugoid Response- Run #2

92
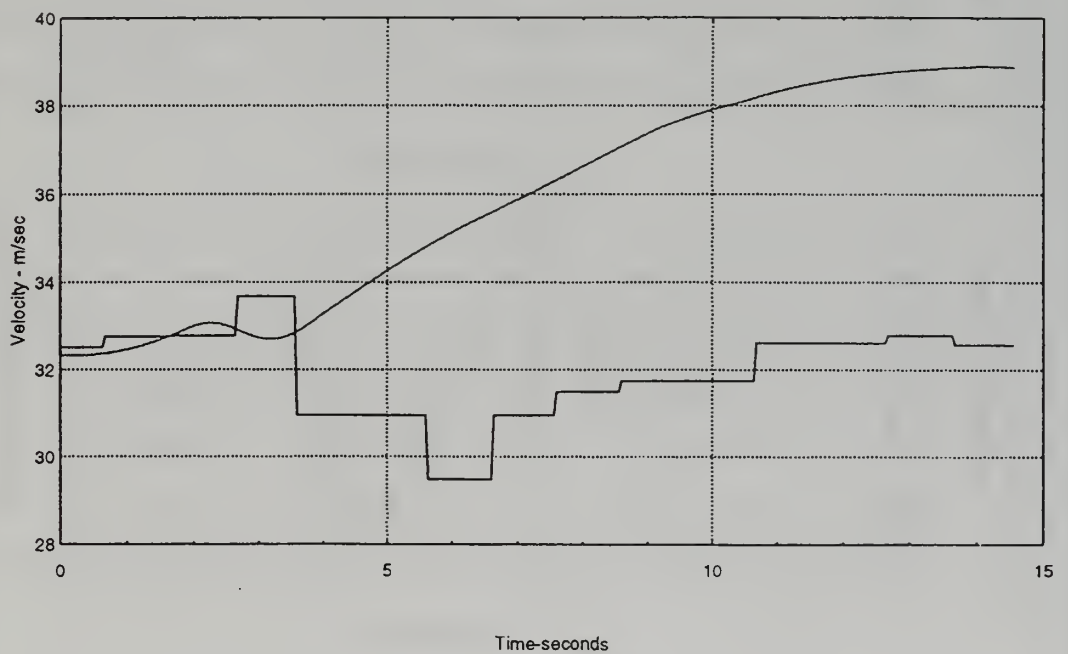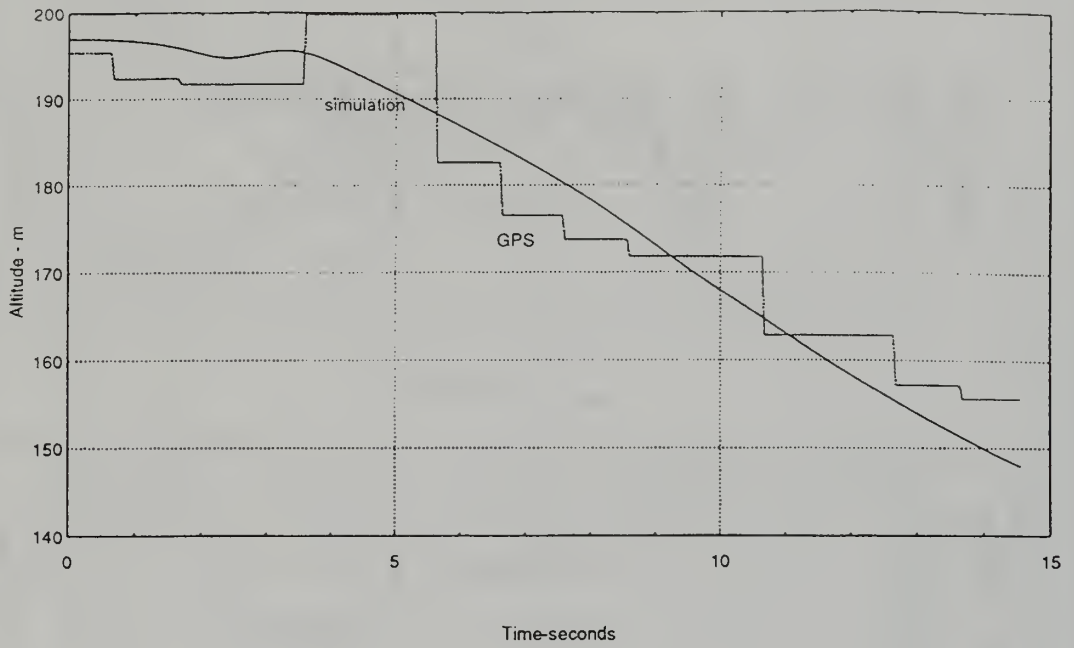
Fig. F.7 Phugoid Response- Run #3

93
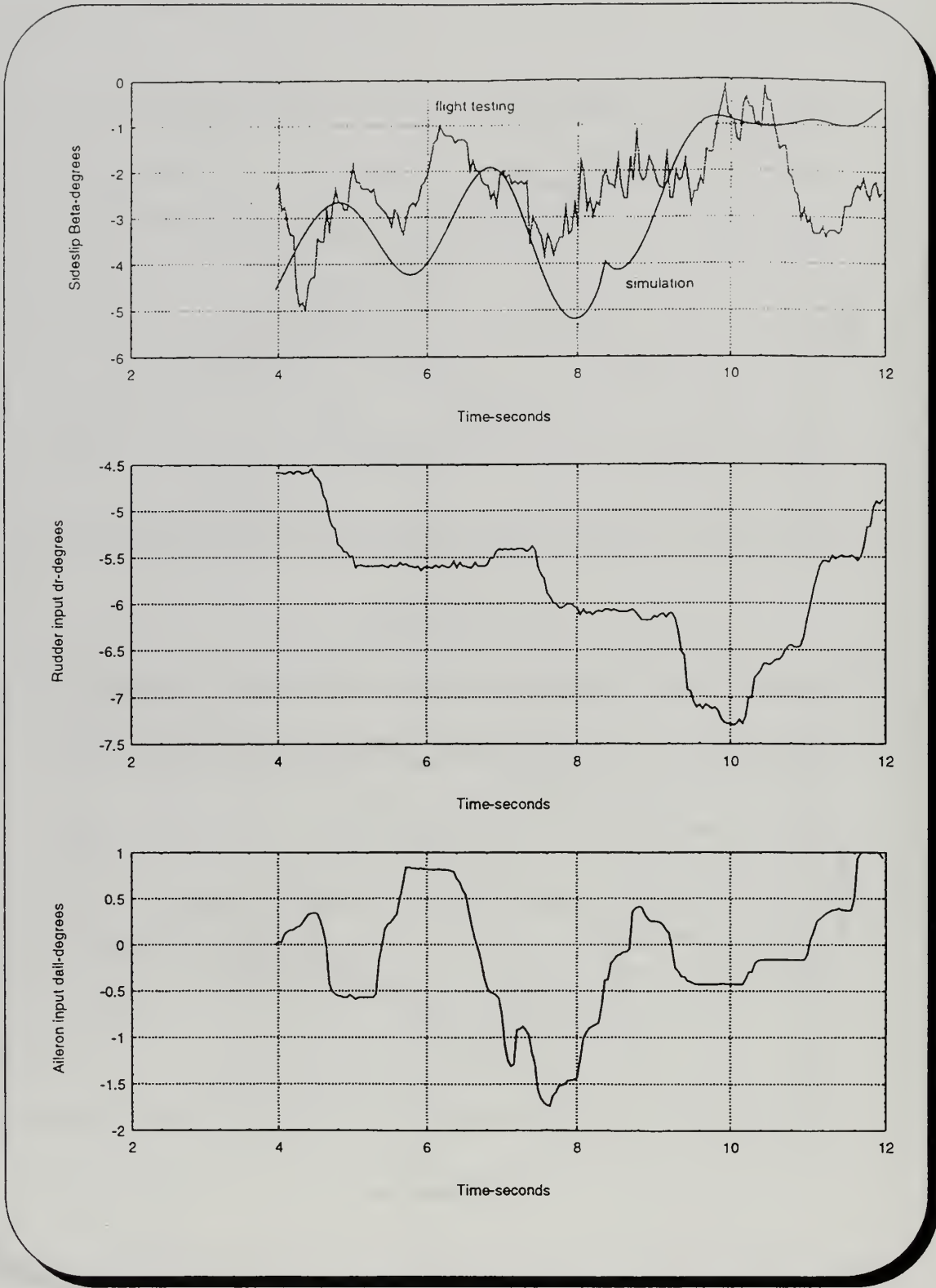
Fig. F.8 Steady Sideslip- Run #1

94
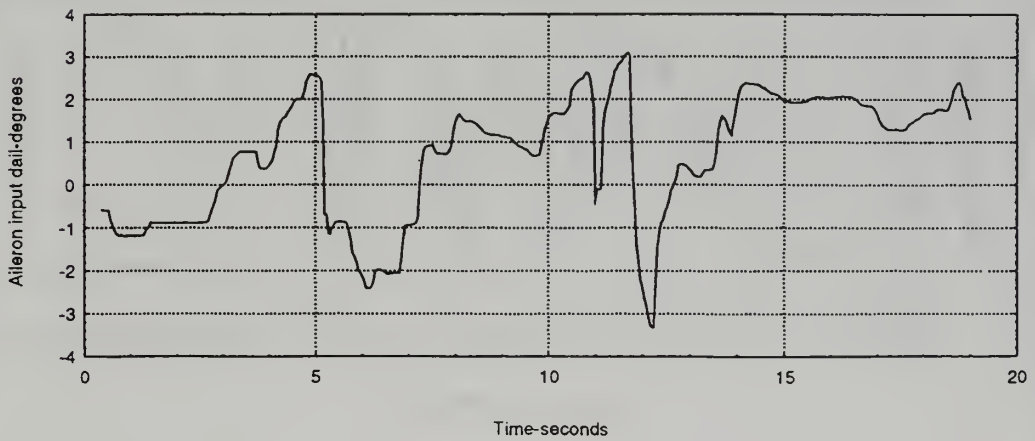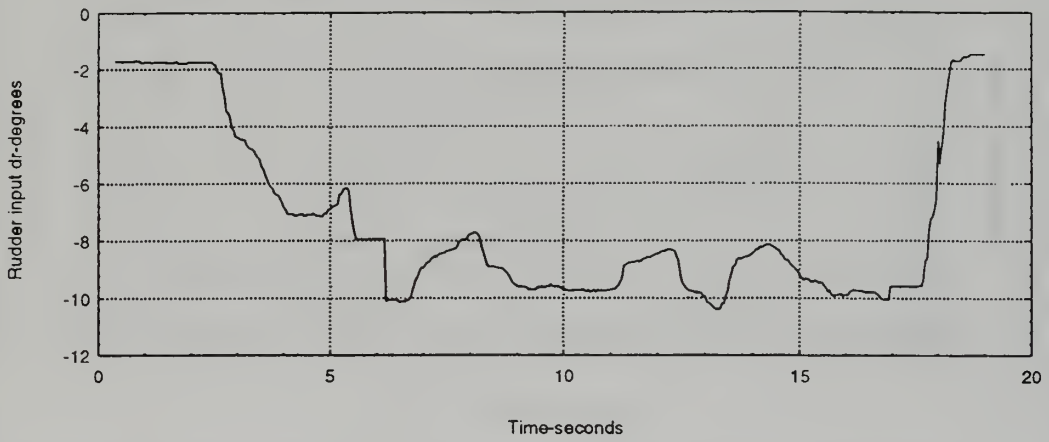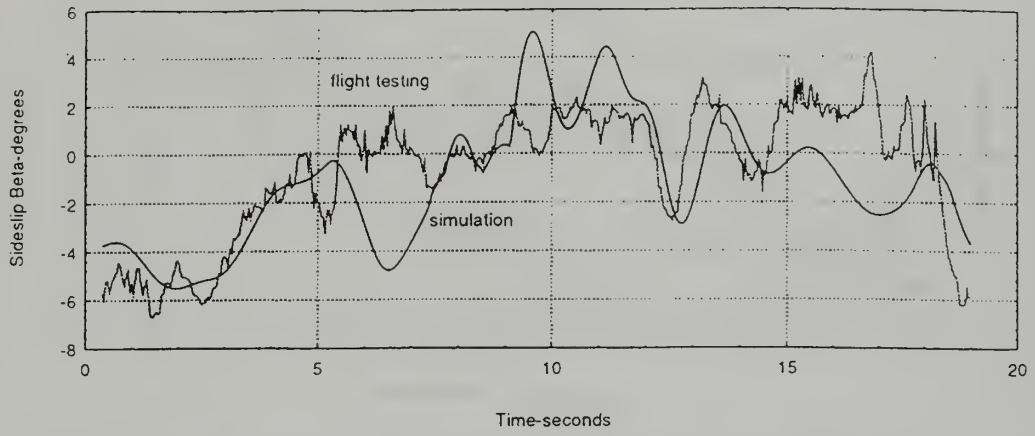
Fig. F.9 Steady Sideslip- Run #2

95
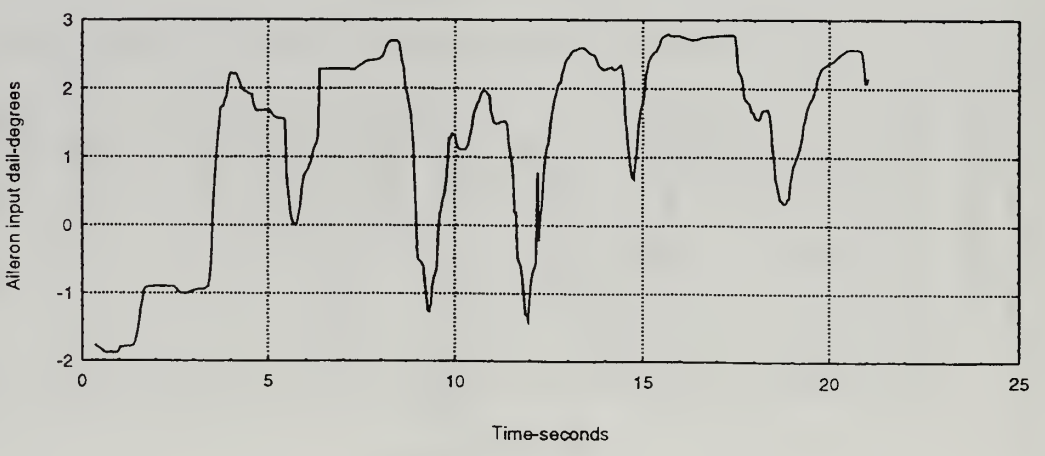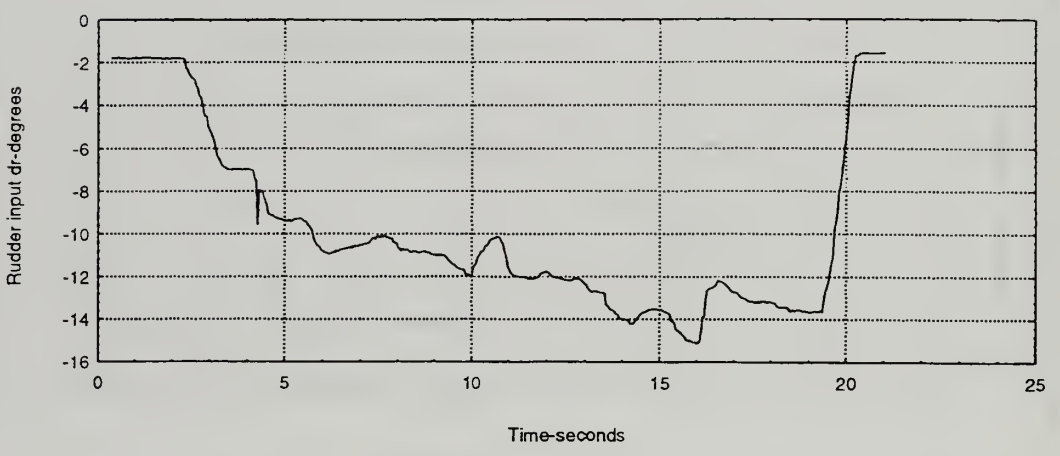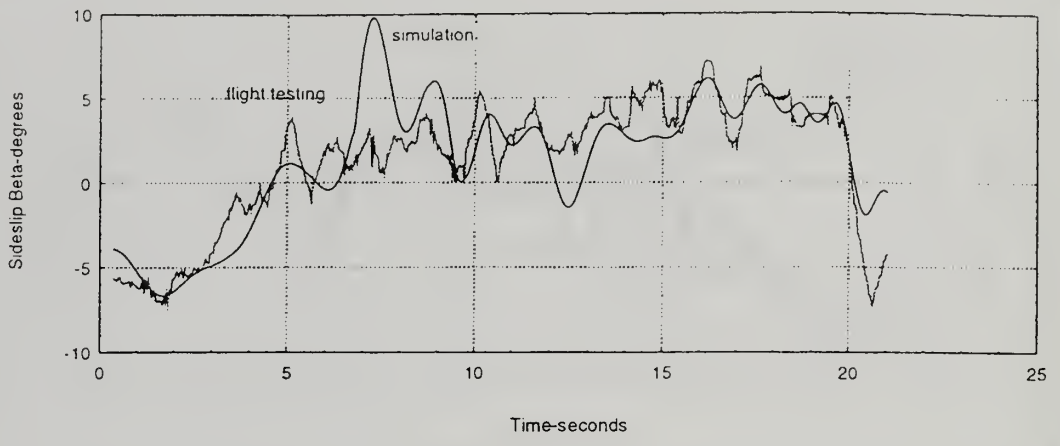
Fig. F.10 Steady Sideslip- Run #3

96
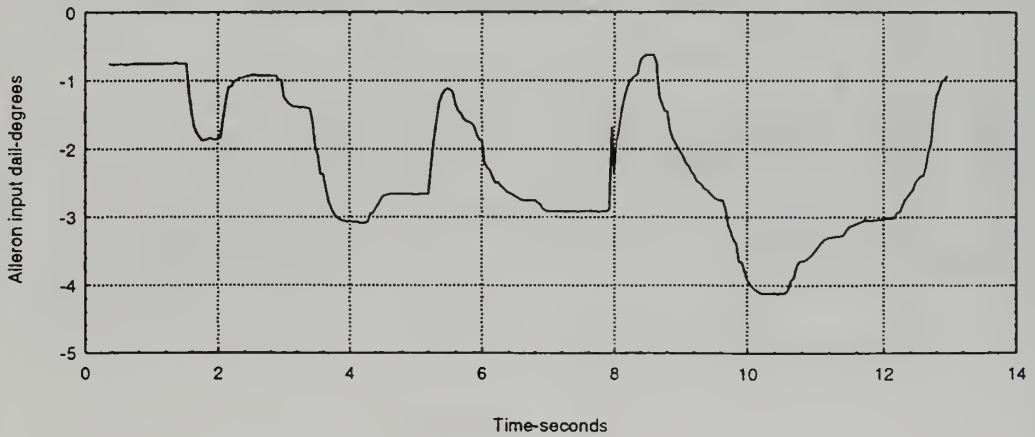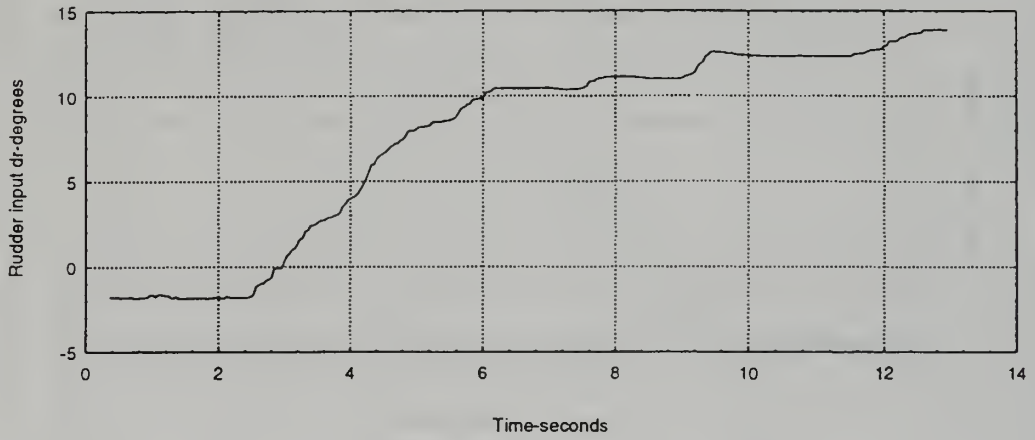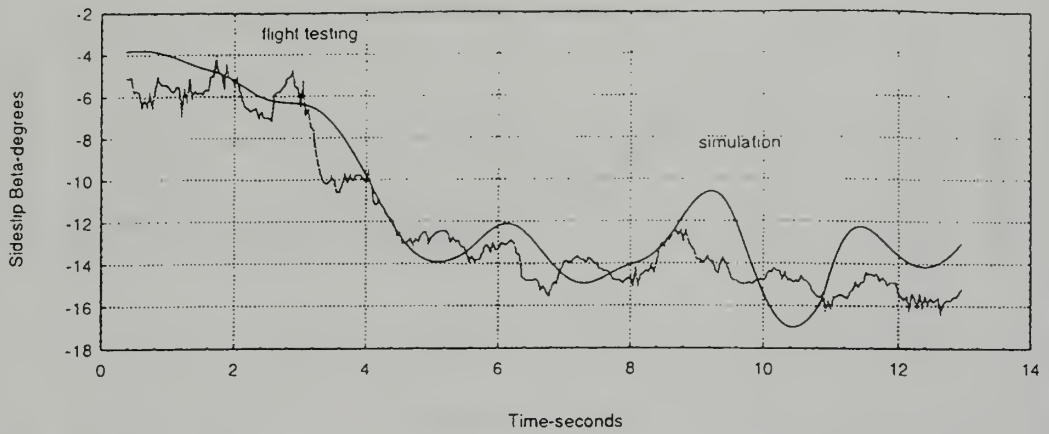
Fig. F.11 Steady Sideslip- Run # 4
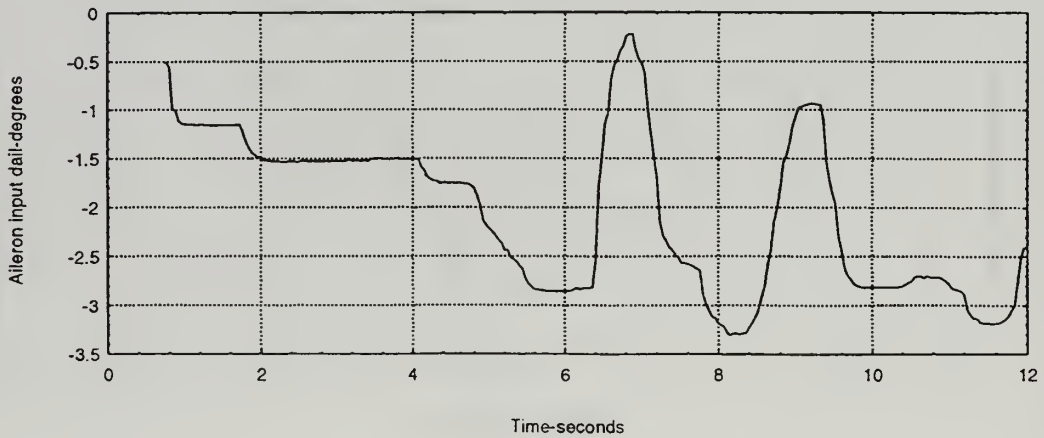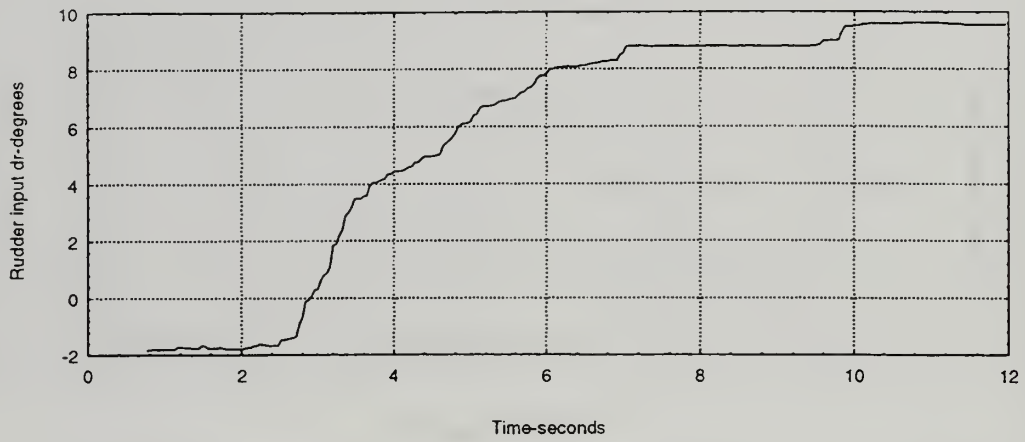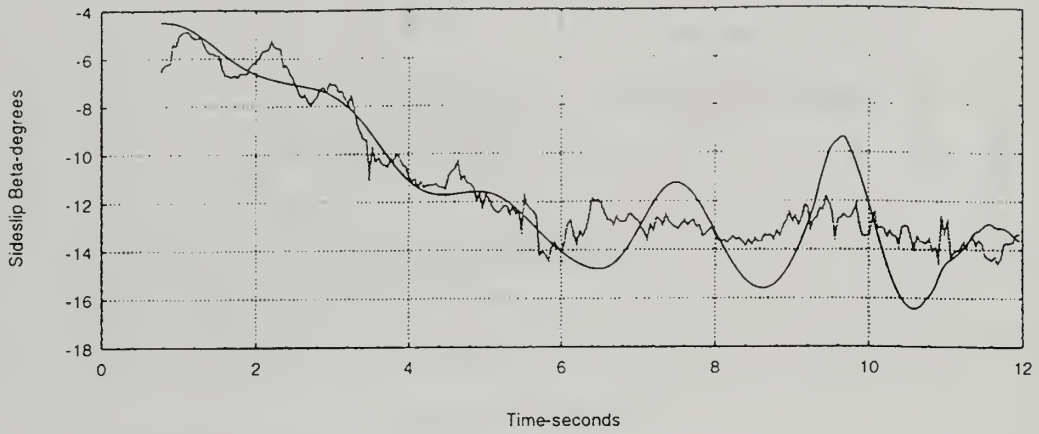
Fig. F.12 Steady Sideslip- Run #5

98
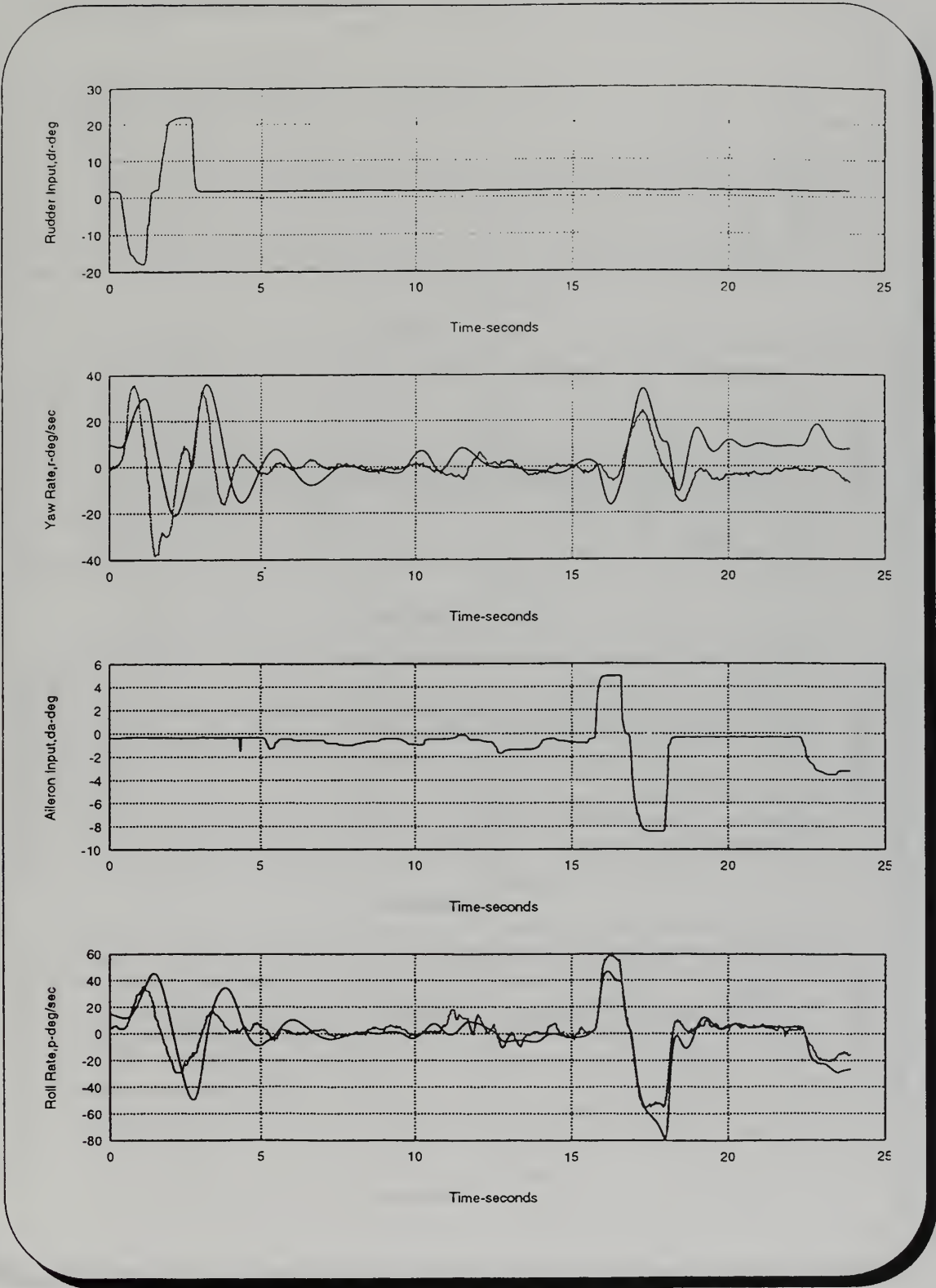
Fig. F.13 Dutch Roll/Spiral Response- Run #1

99
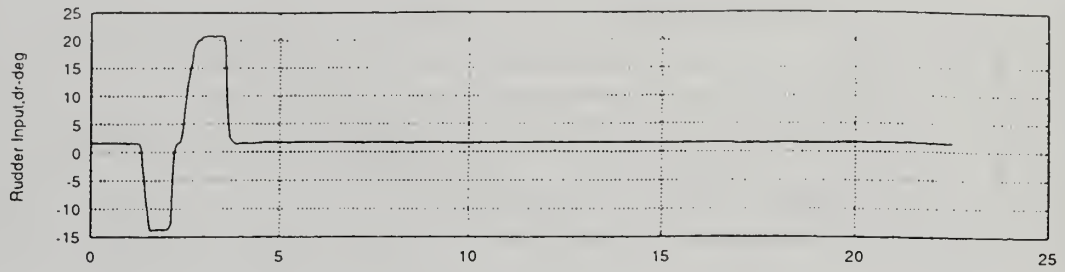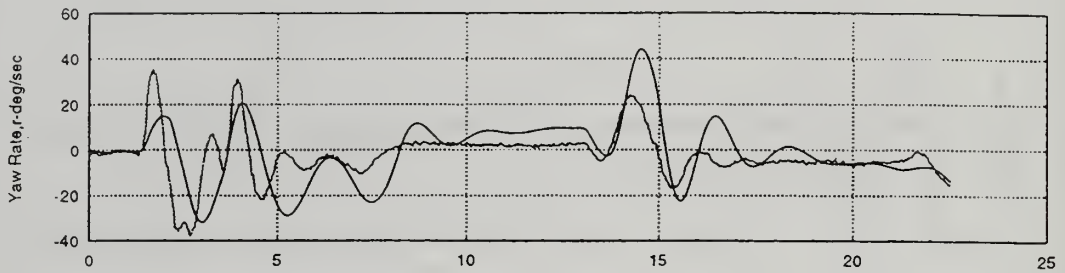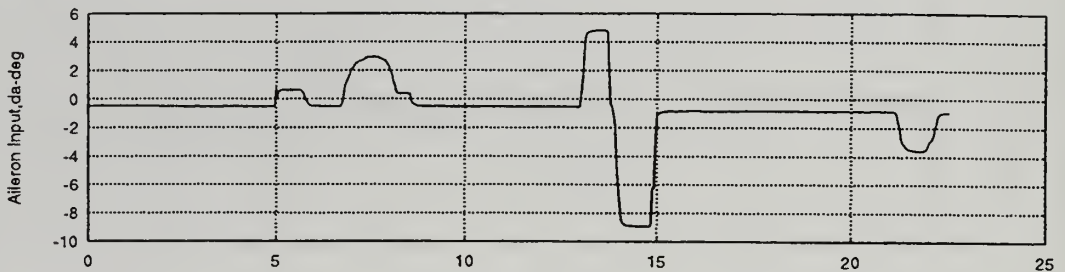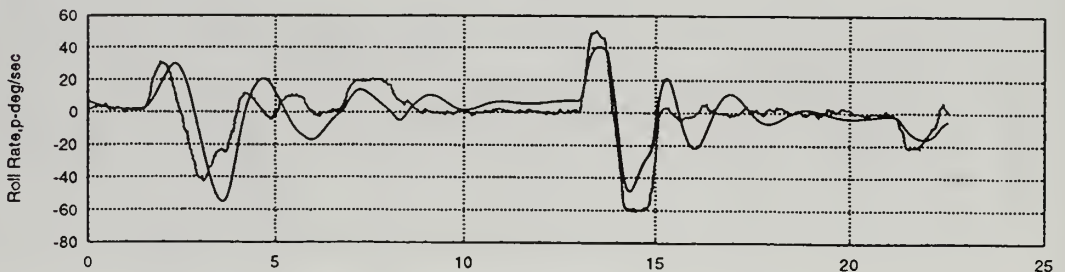
Fig. F.14 Dutch Roll/Spiral Response- Run #2

100
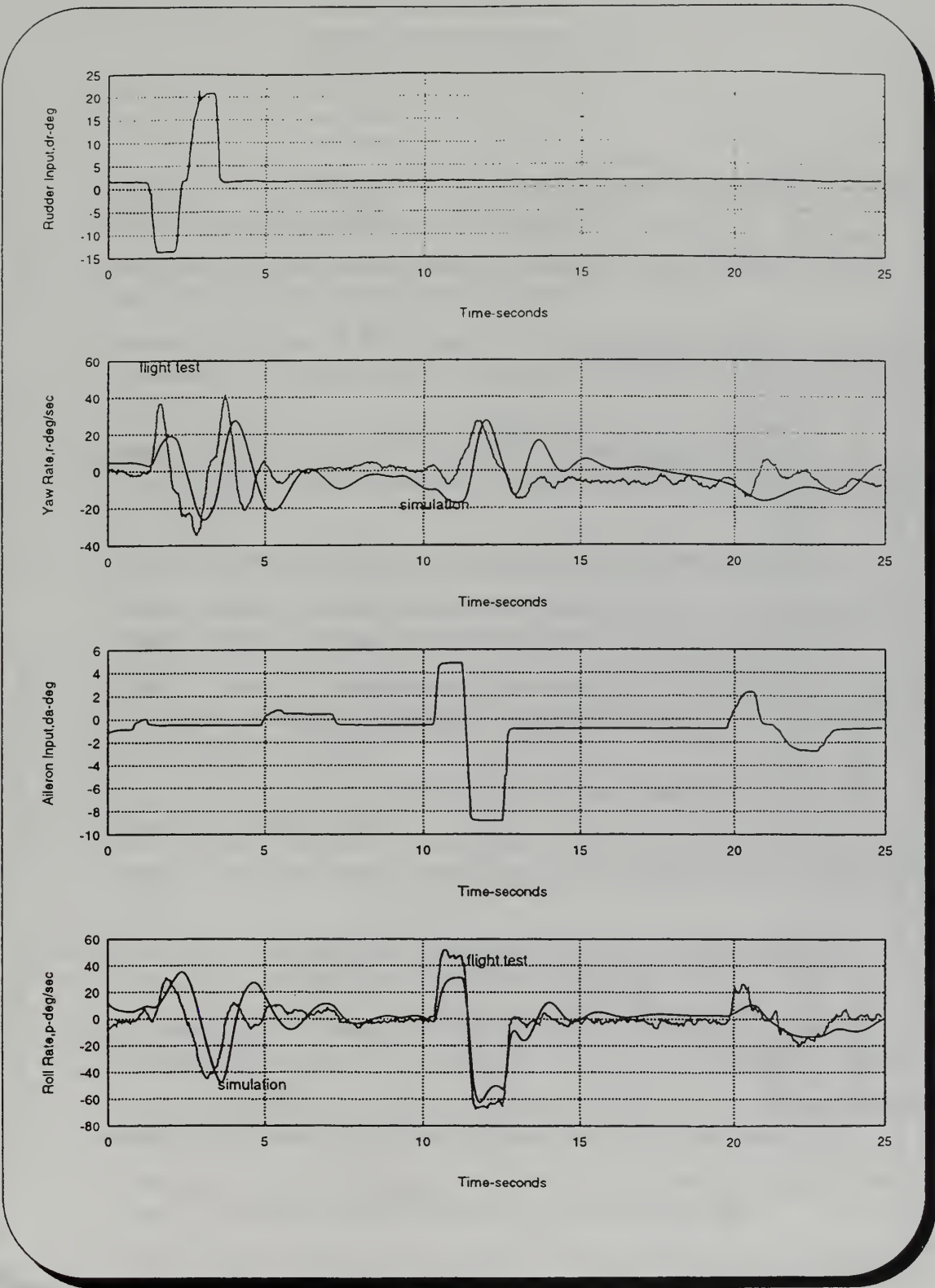
Fig. F.15 Dutch Roll/Spiral Response- Run #3
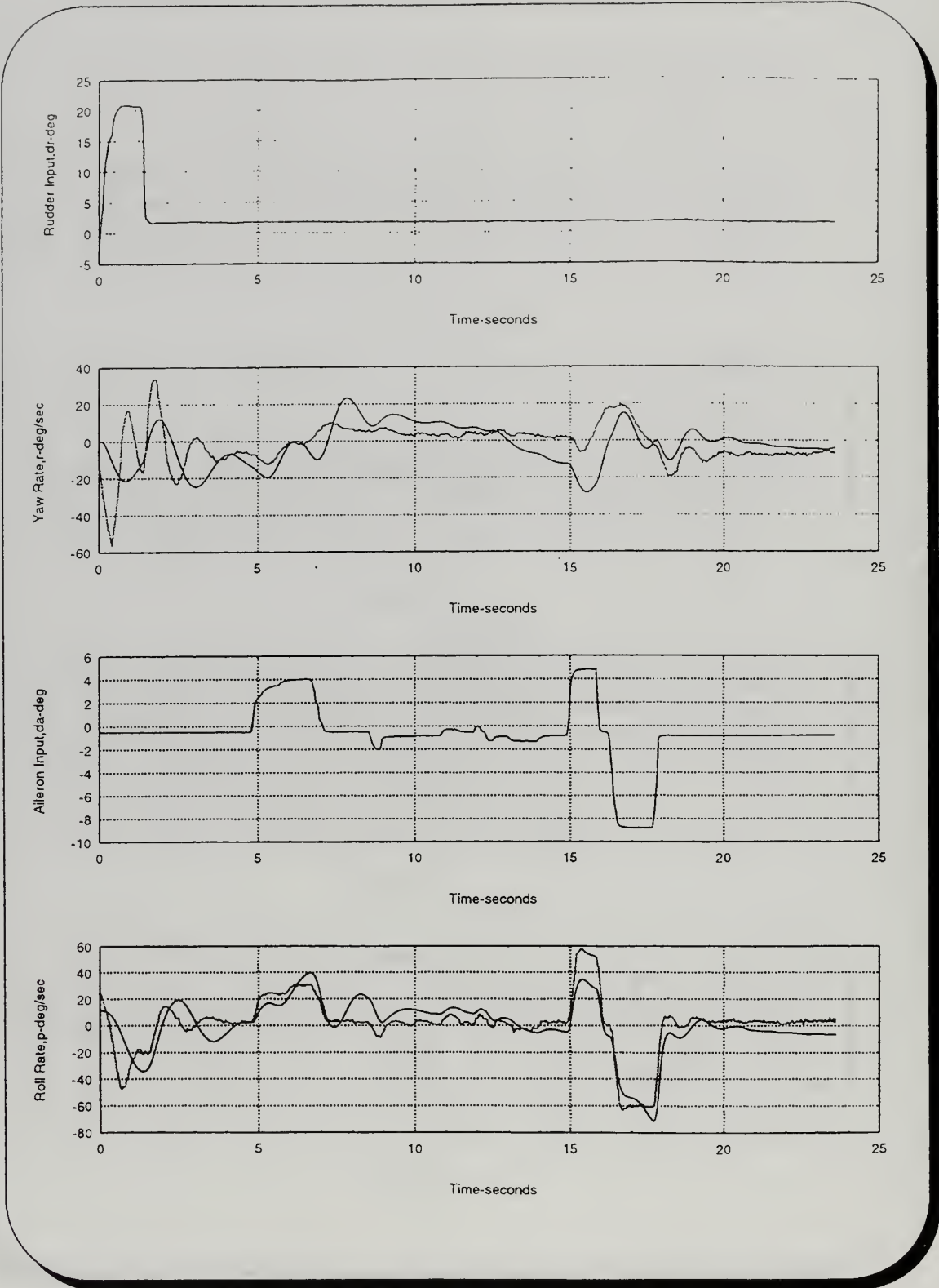
101

Fig. F.16 Dutch Roll/Spiral Response- Run #4

102

LIST OF REFERENCES

1. Philpott, Tom, *Unmanned 'Toy' Played Big Role in U.S. Targeting of Iraqui Forces*, Navy Times, March 11, 1991, pp. 8-11.

2. *Gulf War Experience Sparks Review of RPV Priorities*, Aviation Week & Space Techonology, April 22, 1991, pp. 86-87.

3. Howard, R.M., Bray, R.M., Lyons, D.F., *Flying-Qualities Analysis of An Unmanned Air Vehicle*, Journal of Aircraft, Vol.33, No. 2, March-April 1996, pp. 331-336.

4. Howard, R.M., Tanner, J.C., Lyons, D.F., *Flight Test of a Half-Scale Unmanned Air Vehicle*, Journal of Aircraft, Vol. 28, No. 12, December 1991, pp. 843-848.

5. Hall, Stan, *Dynamic Modelling*, Sport Aviation, July 1987, pp. 30-36.

6. Kuechenmeister, David F., "A Non-linear Simulation for an Autonomous Air Vehicle," Master's Thesis, Naval Postgraduate School, Monterey 1993.

7. Moats, Michael L., "Automation of Hardware-in-the-Loop Testing of Control Systems for Unmanned Air Vehicles," Master's Thesis, Naval Postgraduate School, Monterey 1994.

8. Kaminer, Isaac I., *Class Notes for AA-4276*, Naval Postgraduate School, Monterey, California 1996.

9. Ogata, K., *Modern Control Engineering*, Prentice-Hall, Englewood Cliffs, N.J, 1990.

10. Roskam, J., *Airplane Flight Dynamics and Automatic Flight Controls*, Roskam Aviation and Engineering corp., Ottawa, KS, 1979.

11. Raymer, Daniel P., *Aircraft Design: A Conceptual Approach*, American Institute of Aeronautics and Astronautics, Washington D.C. 1989, pp. 77-83.

12. Smetana, Frederick O., *Computer Assisted Analysis Of Aircraft Performance Stability and Control*, McGraw-Hill Book Company, New York, 1984.

13. Courtland D. Perkins, Robert E. Hage, *Aircraft Performance Stability and Control*, John Wiley & Sons, Princeton, N.J. September 1949.

14. Abbott, Ira H. Von Doenhoff, Albert E., *Theory of Wing Sec*tions, Dover Publications, Inc., New York, 1959, p. 190.

15. Schmidt, Louis V., *Introduction to Flight Mechanics*, Class Notes, AA-3340, Naval Postgraduate School, Monterey, California, July 1992.

16. Howard, Richard M., *Introduction to Flight Mechanics*, Class Notes, AA-3340, Naval Postgraduate School, Monterey, California, January 1996.

17. Maine, Richard E., Iliff, Kenneth W., *Application of Parameter Estimation to Aircraft Stability and Control*, NASA Reference Publication 1168, June 1986.

18. Graham, Robert Gordon, "Implementation of a Personal Computer Based Parameter Estimation Program," Master's Thesis, Naval Postgraduate School, Monterey, California 1992.

19. Zanino, James A., "Uniform System for the Rapid Prototyping and Testing of Controllers for Unmanned Aerial Vehicles," Master's Thesis, Naval Postgraduate School, Monterey, California 1996.

20. Hakun, Mark G., "Operation of an Untethered Unmanned Air Vehicle," Master's Thesis, Naval Postgraduate School, Monterey, California 1995.

21. Watkiss, Eric J., "Flight Dynamics of an Unmanned Aerial Vehicle," Master's Thesis, Naval Postgraduate School, Monterey, California 1994.

22. Beer, Ferdinard P., Johnson, Russell E., Jr., *Vector Mechanics for Engineers, Statics*, Fifth Edition, McGraw-Hill Book Company, New York, 1988.

23. Gainer, Thomas G., Hoffman, Sherwood, *Summary of Transformation Equations and Equations of Motion Used in Free Flight and Wind Tunnel Data Reduction Analysis*, NASA SP-3070, 1972.

24. Howard, Richard M., *Flight Test Engineering*, Class Notes, AA 4323, Naval Postgraduate School, Monterey, California, March 1996.

25. Peters, Mark E., Adrisani, Dominick, *The Determination of Longitudinal Flying Qualities Requirements For Light Weight Unmanned Aircraft*, School of Aeronautics, Purdue University, May 1996.

INITIAL DISTRIBUTION LIST